

大学の理工系学部情報系学科のための
コンピュータサイエンス
教育カリキュラム J97
(第1.1版)

(社)情報処理学会

1999年9月

第 1.1 版 (PDF 形式) のリリースにあたって

1997 年 11 月に公式にリリースした「大学の理工系学部情報系学科のためのコンピュータサイエンス教育カリキュラム J97」の冊子版 (これを「第 1 版」とする) が品切れとなった。これを機会に、同 J97 のマイナーチェンジを図り、第 1.1 版として、今度は電子ファイル (PDF 形式) としてリリースした。これがその第 1.1 版である。

マイナーチェンジに当たっては、情報処理学会が文部省より委嘱された「大学等の情報専門学科における情報処理教育の実態に関する調査研究」(平成 9～10 年度, 委員長: 都倉信樹・大阪大学教授) の活動をもとにしている。この調査研究の平成 10 年度報告書 (1999 年 3 月) も情報処理学会 (<http://www.ipsj.or.jp>) より入手できる。

第 1.1 版は第 1 版のマイナーチェンジであり、誤字や脱字の訂正と第 1 版出版後に入手した情報をもとにした詳細シラバスの更新である。したがって、J97 の設計思想、及び、講義科目やモデル履修コースの設定等には変更はない。

誤字や脱字の訂正以外の主な変更点は、以下の各科目の詳細シラバスの更新である。

- L-2 プログラミング入門；
- M-3 計算論；M-4 確率論と情報理論；
- U-3 データ構造とアルゴリズム；U-5 プログラミング言語論；U-8 コンパイラ；
- U-9 デジタル通信；U-12 情報ネットワーク；U-16 信号処理；U-18 パターン認識；
- A-7 信号処理特論；

1999 年 9 月

柴山 潔 (J97 策定ワーキンググループ・主査)

大学の理工系学部情報系学科のための コンピュータサイエンス教育カリキュラム J97

J97 策定ワーキンググループ (J97-WG)
(社) 情報処理学会・情報処理教育カリキュラム調査委員会

1997年11月

1 はじめに

大学におけるコンピュータサイエンス(本報告書では, 情報工学, 情報科学, 計算機科学, 計算機工学などの総称として「コンピュータサイエンス(以降“CS”という)」を使う)教育のカリキュラムの検討は国内外の学会の主導下で盛んに行われ^{[1][6][8]}, それらによる提言に従った大学教育カリキュラムの改訂や再編が各所で行われてきた。例えば, 国外では, ACMが1968, 78, 88, 91年の4回に渡り, CS教育カリキュラムの改訂^{[5][6][7]}を行っている。一方, 国内では, 文部省の委託を受けて情報処理学会が“J90”と呼ぶ大学の情報系学科におけるCSの専門教育コアカリキュラムを提言^{[1]~[4]}している。

しかし, 情報処理技術の進展とともに, 社会や産業構造が転換し, 情報処理技術を支えるCS分野が学問分野としても成熟した点を受けて, その教育カリキュラムのなお一層の充実・整備が急がれている。

国内では, 情報処理学会に設置されている情報処理教育カリキュラム調査委員会(現委員長: 都倉信樹・大阪大学教授)が, J90の提言以降も, 大学の理工系学部情報系学科の専門教育カリキュラムについて精力的に検討を続けている。そして, この情報処理教育カリキュラム調査委員会の主導で, J90提言以降のCSの発展を踏まえた新しいCS教育カリキュラム(これを“J97”と称する)を策定することになり, その策定作業を行うワーキンググループ(主査: 柴山潔・京都工芸繊維大学教授)(以降, “J97-WG”という)が組織された。J97-WGは, 1994年8月より作業を開始し, この1997年9月に3年に渡る策定作業を完了し, ここに情報処理学会より公式に「大学の理工系学

部情報系学科のためのコンピュータサイエンス教育カリキュラム J97」として出版されることになった。

J97-WGの委員(付録I参照)は, i) 情報系専門学科の卒業生自身で新しいカリキュラムを作成する; ii) できるだけ幅広い専門分野をカバーする; などを要件に選定された。結果として, J97-WGは13名の情報系学科卒業生から構成(その他に, WG委員には都倉・情報処理教育カリキュラム調査委員会委員長も含む)されている。

本報告書では, 情報工学, 情報科学, 計算機科学, 計算機工学などの総称であるCSを教育対象とする大学の理工系学部の情報系学科のための新しいCS教育カリキュラム J97(詳細は付録III参照)を提示する。

2 大学の理工系学部情報系専門学科の役割

本章では, 大学における専門教育, さらには, CS教育の意義や意味(位置付け)について考察を加える。

2.1 大学教育の役割

2.1.1 大学教育課程とその入出力

本報告書で議論の対象とする大学教育とは, 4年間の学部教育を中心に考える。従って, 大学における学部教育の在り方の検討では,

- 大学教育課程への「入力」である初中等教育(特に, 高校や高専の教育)の内容, すなわち大学での専門教育を履修するために入学前に必要な知識;

- 大学教育課程の「出力」である社会及び産業界の期待や要求、すなわち大学卒業後の進路；の2点について十分に考慮することが必要である。

一方、大学教育の現場においては、近年の大学改革の嵐の中で、具体的には、i) 一般教育課程の再編；ii) 大学院の重点化；iii) 社会人/職業人/一般人の再教育（リフレッシュ/リカレント教育）の励行；iv) 入学資格の門戸拡大（高専/短大卒業生の編入など）；v) 自己点検や外部評価；vi) シラバスの整備；などが実施されている。

このうち、i)～iv) が教育カリキュラムに与える影響は大きく^[9]、当然のことながら、大学改革では教育カリキュラムの抜本的な改革が目玉あるいは必須となっている。

具体的には、

- 入学者の多様化による専門教育に必要な前提知識（先修知識）の多様化や未履修者の入学；
- 編入や飛び級による専門教育期間の多様化や短縮；
- 一般教育と専門教育とが融合される中での、専門教育の本質の確保；

などによって、「画一的ではなく多様化に対応できる柔軟性を備えた専門教育カリキュラム」が要求される。専門教育カリキュラムの設計においては、この「多様性への配慮と専門的知識修得の実現とのトレードオフ調停」が困難ではあるが克服すべき課題となる。

2.1.2 初中等教育との関係

大学の学部専門教育課程への入学者（入力）は多様化したといえども、まだ依然として高校や高専の卒業者が多数を占めている。学部専門教育と初中等教育との関係^{[12][13]}では、次の2点が、専門教育カリキュラム設計において考慮すべき具体的な影響として考えられる。

- 入試科目の減少などによって、専門教育に必要な前提知識（例えば、数学や物理など）の未修得者が入学する。
- 初中等教育における「情報」科目の設置による情報やコンピュータのリテラシ教育と大学におけるそれとの連携・統合が必要となっている。

いずれも、初中等教育で修得してくる知識と大学の専門教育が要求する前提知識との不整合に起因する問題点である。この不整合が著しくなる

と、結果として、大学専門教育課程での入力条件の整備（前提知識の教育のやり直し）が必要となり、i) 専門教育時間の圧迫；ii) 専門教育のレベルダウンを伴う一般化；などの悪影響が生じる。

また、特に後者の点では、「初中等学校だけではなく、情報系専門学科以外の大学も含めた教育現場でのコンピュータリテラシあるいは情報リテラシ教育がCS専門教育カリキュラムから受ける影響は大きい」ことを考えて、CS専門教育カリキュラムの策定においても、両者の関係や分担についても十分配慮する必要がある。

2.1.3 社会や産業界との関連

近年、大学を取り巻く社会や産業界が構造的な転換点を迎えている。「社会や産業構造の転換」とは、具体的には、次の(1)(2)の点であり、それがCS教育に及ぼした影響について考えてみる。

(1) 情報インフラの整備

「科学技術創造立国」を謳う科学技術基本法の施行（1995年11月）及びそれに基づく科学技術基本計画の閣議決定（1996年7月）などによって、情報インフラ（基盤；infrastructure）の整備事業が重点的に予算化された。これは既存の情報処理産業の高度化を促している。このため、それらの産業に人材を供給する大学のCS専門教育の高度化・先端技術教育化をもたらしている。

(2) 新しい社会現象や情報産業の発現

コンピュータやインターネット利用の爆発的な普及は社会的・文化的にも様々な影響力を行使し、それによって生み出された諸概念（例えば、仮想商店街、電子取引、電子出版、仮想社会など）が情報流通をもとにする新しい情報産業を創出している。これらは、CS分野の学問及び情報処理技術のブレークスルーと持ちつ持たれつの関係であり、それらの社会現象や産業を支える職業人（技術者、芸術家、教員など）に基礎知識を教授するCS教育が求められている。

「（専門教育を受けた）専門家」とは「ある技術のスペシャリスト（specialist）」であり、その育成が大学の専門教育においても求められる。特に、CSは幅広い広がりを持つ学問分野となっており、CSの一般的な（広いが浅い）知識の修得（ジェネラリスト（generalist）の養成）だけでは専門教育にはなり得ないことに注意する必要がある。「色々

なことを知っているけれども、深くを語れる知識をひとつも持たない」ではなく、「1分野でよいかから専門分野に関しては、どこもブラックボックス化せずに語れる」人材を職業人として社会や産業界に輩出することが必要である。

2.1.4 大学院との関係

2.1.2項で述べた大学専門課程の入力条件の変化は出力にも影響を及ぼすのは当然である。

学部の卒業生(出力)の受け皿としては、社会や産業界だけではなく大学院も存在する。そして、入力条件の変化の影響、及び2.1.1項で述べた大学改革の実例のうちii)の大学院の重点化によって、実際には、

- 「社会で通用する知識を修得した」という自信を持つために大学院進学を希望する(一種のモラトリアム)学生が増えている；
- 産業界が開発現場の即戦力にならない(と思っている)学部卒業生の採用を避ける；

などの現象が目立ってきている。

このため、学部の教育カリキュラムも学部だけの閉じたカリキュラムではなく、大学院も視野に入れた、さらには「学部教育と大学院(特に、博士前期課程)教育との連係を考慮した専門教育カリキュラムの編成」が必要となっている。例えば、学部-大学院の6年間の専門教育カリキュラムの実現によって、初めて、専門教育課程の修了者(出力)と産業界の要求する人材との整合を図ることができる場合もある。

2.1.5 大学における教育と研究

社会や産業界に直結する高等教育機関としての大学(大学院も含んで良い)における教育(すなわち「一般性の追求」と研究(すなわち「独創性の追求」)にはトレードオフがあり、大学における教育カリキュラム構成に重要な影響を及ぼす。

ほとんどの大学の理工系学部には、教育と研究の両方の性質を兼ね備えるものとして卒業研究がある。卒業研究は、教育面からは「教員と学生との1対1の個人的実験/演習」と、研究面からは「研究者としての第一歩」と、それぞれ位置付けられる。卒業研究は教育カリキュラムの最後に位置し、「専門教育課程のまとめを行う」という重要な役割を帯びている。従って、2.1.1項で触れたよ

うないいわゆる飛び級やリカレント/リフレッシュ教育における「卒業研究の未履修/免除」の影響は大きく、専門教育カリキュラムにおけるこれへの対策は必要である。

2.2 CS教育の特質

2.2.1 学問体系としてのCSの位置付け

CSの教育原理を整理すると、次のような2つの大きな特徴が浮かび上がる。

- (1)「『情報』という論理的概念の取り扱い方」として構築した学問体系を種々の機能レベルで実証する必要がある。
- (2)情報処理における抽象化レベルによってできる情報処理の各階層をCSの各教育分野に対応させられる場合が多い。

(1)の特徴はCS教育の本質(原理)が「論理的概念と物理的事象との対応付け(mapping)」にあることを示している。CSでは「モデル化」「形式化」「抽象化」あるいは「仮想化」などと呼ばれるこの対応付けによって、論理的概念あるいは枠組みを実証する必要がある。実証方法としては、ハードウェアという物理構造での実験からソフトウェアによる抽象構造の実現まで種々の対応付けレベルが考えられる。また、他の学問分野に比べてCSでは、この論理的構造と物理的構造の対応付けの適用可能性が高くかつ本質的となる。この意味で、「CSは実証科学(科学と工学との融合)であり、物(ハードウェアやソフトウェア)作りが本質的な意味を持つ」と言うことができる。

次に、(2)の特徴は「情報処理の各機能階層がCSにおける重要な個々の学問領域に対応し、それらが相互に階層的に関連すること」に見られる。すなわち、情報の抽象化の度合い(情報処理階層)の各々に対応する学問対象(学問領域)が存在し、それらが情報処理階層に従う連続的な上下関係(包含関係)で密接に関連付けられる場合が多々見られる。たとえば、CS分野での研究を支えるあるいはその対象そのものとなる情報処理システムやコンピュータシステムの機能階層関係において見られる「コンパイラ、オペレーティングシステム、プログラミング言語とコンピュータアーキテクチャとの密接な関連」などは「抽象化レベルの上位概念の教育に下位概念の知識が必要となる場合がある」の実例である。

また、(2)の特徴は「ある学問領域の自己組織化の可能性を引き出す」という利点も生み出しており、それがそのままCS教育の困難さにも直結している。各分野を「点」として独立・排他的に教育するのではなく、それらを「線」や「面」として結び付け、関連性や一貫性を重視して、また総合(統合)的に教育する必要がある。また、教育効率や種々の制約(人材や時間、環境など)を考慮すると、「ある学問領域を教育する際にそれより下位レベルの階層を隠蔽(ブラックボックス化)する必要」が生じてくる。それとともに、このブラックボックス化によって教育対象の欠落が生じてはいけない。CSでは、ある学問分野の教育に際し、「その科目を学ぶ際に最小限必要な知識(先修科目)は何か?」を見きわめ、下位概念の適切なブラックボックス化を図る必要がある。

2.2.2 CSの教育目標

前の2.2.1項で述べた(1)(2)の2つの特徴をまとめ整理すると、次のような大学におけるCS専門教育の具体的な教育目標が導き出される。

CSでは、「計算(computation)を理論の対象にするためのモデル化、抽象化、仮想化、離散化」を行う。この点から導き出される「CSの教育目標」とは、次の3点である。

- 計算や情報処理に関する単なるノウハウではなくて系統化された知識体系を教える。
- 計算(情報処理)に対する総括的かつ柔軟な見方ができて、メタ(高次)レベルの計算体系を考えられる学生を育てる。
- 解が1つでない問題に対処できる、すなわち複数の選択肢(alternatives)を列挙し、比較・評価(吟味)する能力を養う。

すなわち、CS教育においては、学生に対して、論理 ⇔ 物理、モデル(形式) ⇔ インプリメント(implementation; 実現)、理論 ⇔ 実験(実証)、理想 ⇔ 現実などの対応付けやその機能差を埋める種々の体系的方法を論じる素養を身につけさせる必要がある。

また、一方で、CSは「計算機構の直接的実現(インプリメント)」も教育原理の1つとする必要がある。これは、具体的には、「モデル化した計算機構の実装・評価」と考えられる。この点からは、次の2点のCS教育目標が導き出せる。

- CS分野の学問は、解析(analysis)を重視する科学(science)と合成(synthesis)を重視する工学(engineering)とが融合している。従って、CS分野では、解析だけではなく、システム化や合成の能力も必要となる。すなわち、ハードウェアとソフトウェアを組み合わせる情報処理システムを構築する能力が必須である。
- CS教育においては、ブラックボックス化が鍵となり、またその機能の決定は慎重に行う必要がある。すなわち、「仮想化あるいは抽象化技術を駆使する方法論を修得/体得させる」もCS教育の重要な目標となる。

すなわち、「CS」は、厳密にいうと「コンピュータサイエンス(科学)(CS) + コンピュータエンジニアリング(工学)(CE)」すなわち「CS & CE」である。

2.2.3 CSにおける教育と研究

CSは、2.2.2項で述べたように、科学(理学)と工学の両者の学問的特質を兼ね備えており、その意味で、CS教育がカバーすべき知識体系は非常に多岐に渡るのが特徴的である。また、CSの専門学科の卒業生の代表的な進路は先端科学産業界であり、そこで要求される知識や技術と大学の専門教育で学習するCS一般の基本的知識との隔たりが広がりつつある。CSの特徴の具体例とその専門教育との関係については、3章でJ97の設計方針を明示する際に指摘する。

大学においては、研究と専門教育との関係は深い。具体的には、ちっぽけでも独創的な物(CSにおける例:プログラミング言語;コンパイラ;オペレーティングシステム;コンピュータアーキテクチャなど)を作る能力を養う教育が必要であり、物作りの実験/演習に補完された教育が研究につながる。言い換えると、教育の観点を失うと研究はできない。教育の原理や思想が見えることが重要であり、それが独創的な知識の普及、宣伝、名前付け(ネーミング)につながり、研究のシーズとなる。

2.2.4 CS教育カリキュラムの編成方法

教育カリキュラムの編成時には、2.2.1項の最後で述べた下位概念のブラックボックス化は先修科目(ある科目の講義に先立って講述しておいた

方が良い科目)との関連で注意を要する。すなわち、CS教育カリキュラムの構成時に問題となるのは、情報処理の機能階層各々に対応付けた科目の教育順序(先修科目)の決定である。情報処理の機能階層は抽象化レベルであり、またそれは下位階層のブラックボックス化であるので、各階層に対応する科目を学年進行に従って、a) ボトムアップ(下位から上位へ)に履修する；b) トップダウン(上位から下位へ)に履修する；c) a)b)の両者を混在させる；の3通りのカリキュラム編成が考えられる。

例えば、初期(1970年代)のCS教育においては、a)のボトムアップ編成のカリキュラムが多く見られた。ソフトウェア科学/技術の未成熟と「はじめにハードウェアがある」の考えが教員側の底流にあったことが最大の理由であろう。

しかし、ハードウェア技術だけではなくソフトウェア科学/技術の劇的な進展・拡大に従って、b)のトップダウン編成カリキュラムの重要性が見直されるようになった。例えば、J90^[1]では、一部にトップダウン編成の考え方を取り入れて先修科目を規定している。ただ、種々のトレードオフについて論じる素養の教授を目指すCS教育においては、a)b)の両編成法のいずれかだけに頼ってカリキュラムを構成するのは無理がある。J90においても、「コアカリキュラムの前半部分(現在のコンピュータリテラシに当たる部分)はトップダウン編成であるが、後半部分(特に、コンピュータシステム分野からコンピュータ応用分野にかけて)はボトムアップ編成する」というc)の方法になっている。

2.2.5 CSの発展とその教育への影響

J90以降にも学問分野としてのCSの発展は著しい。特に、その専門教育に多大な影響を及ぼしている事項を列挙すると次のようになる。

- (1) ソフトウェア科学/技術の比重が大きくなった。
- (2) コンピュータシステムの応用/適用分野が著しく拡大した。
- (3) コンピュータあるいは情報インフラの整備がハードウェア、ソフトウェアともに進んだ。
- (4) ヒューマン-マシン・インタフェースを中心とする「柔らかな技術」が重要となった。

また、CS教育そのものの様変わりも著しい。特に、次の2点は重要な事項である。

- (5) コンピュータリテラシあるいは情報リテラシの教育環境が充実してきた。
- (6) CS学科が増大し、その教員や教育環境が充実してきた。

列挙したCSの変遷がCS教育に及ぼした具体的な影響について考察してみよう。

まず、(1)の点では、「メインフレーム中心からマイクロプロセッサ中心へ」が背景にあり、マイクロプロセッサをコンピュータシステム構築用の高機能部品として活用することによるコンピュータシステムの多様性が、CS(特に、ソフトウェア系分野)教育の守備範囲を拡大している。

(2)の点では、「CSの学際領域への進出」があり、CS教育でカバーすべき範囲も拡大・多様化の一途をたどっている。それが教育方法や環境に与える影響も大きい。特に、CS分野の拡大によって、コンピュータの応用分野の科目が増大している。

(3)の点は、具体的に、分散化、ダウンサイジング、パーソナル化、台数の拡大、マルチメディア化、ネットワーク/通信の高速化/高機能化、インターネットの活用などを指し、そのために必要な技術の基盤(インフラ)作りがCS教育に求められている。

(4)は、コンピュータ側に立つ技術あるいはコンピュータ機能だけを考えた技術ではなく、人間の心理や生理に踏み込んだ「ヒューマンインタフェース」あるいは「ヒューマン-コンピュータ・インタラクション」などの講義科目の重要性を示している。

また、(5)(6)の点では、コンピュータユーザ層の拡大(低年齢化と高年齢化)がCS教育カリキュラムの多様化を促している。また、これらによって「コンピュータリテラシ教育とCS教育とが完全に分離独立して扱えるようになった」ことがCS専門教育の設計に与える影響も大きい。

2.2.6 社会から期待されるCS教育

大学教育課程の出力である社会の期待や要求すなわち大学卒業後の進路のうち、特に、CSを修めた人間が就くことの期待できる職業としては、i) システムエンジニア(SE)；ii) 基本ソフトウェア開発者；iii) 応用ソフトウェア開発者；iv) シス

テム LSI 開発者；v) ハードウェアシステム開発者；vi) 大学における CS 専門教育，産業界等における情報技術 (IT: information technology) 教育，さらには，初中等学校における情報教育などの各教員；などがある。

しかし，例えば，i) の SE という職業でも，2.2.2 項で述べた CS の学問的本質である「数ある選択肢を列挙し，それらを比較検討・評価して，対象に対して最適な選択肢を選ぶ能力」あるいは「モデル化 物作り 評価/最適化のすべての過程をこなすことができる能力」を修得していることが肝要である。特に，バブル期に増産された SE には，「物作り」能力だけを問われて，「モデル化」や「評価/最適化」の能力が欠如している場合が多く見られた。また，1980 年代以降に生じたいわゆる「ダウンサイジング」では，それまでのメインフレーム全盛からマイクロプロセッサへ SE の対象となるコンピュータアーキテクチャが激変している。メインフレームの専門家であっても，マイクロプロセッサという別解に目を向けられない SE では，社会情勢の激変や産業構造の転換に耐えられないのである。

iv) のシステム LSI 開発者とは，いわゆる半導体メーカーの SE であり，現在の半導体産業が最も渴望している^[10] 職種である。この職種に要求される能力は「ハードウェアとソフトウェアの両方を視野に入れ，それらのトレードオフを適切に決定することのできる技能」である。まさに，CS の学問的本質に最適な職種であり，CS 専門教育課程の出力として考慮すべき産業であろう。

vi) については，2.1.1 項及び 2.1.2 項でも述べたように，初中等学校に「情報」科目が新設され，初中等教育における「情報処理 (情報リテラシ，コンピュータリテラシ)」教育の重要性が指摘されている^{[11][12][13]} 背景がある。特に，それらを担当する教員不足という深刻な現状があり，これを是正するためには，CS 専門教育を修めた者を初中等学校における情報リテラシ及びコンピュータリテラシ教育者，あるいはその養成機関の教員として就ける道を開く」というような抜本的な対策も必要である。CS 専門教育課程の出力 (卒業/修了者) をその入力条件 (初中等学校における情報リテラシ教育) の整備に活用するのである。

3 CS 教育カリキュラム J97 の設計

本章では，2 章でのさまざまな観点からの考察をもとにして策定した新しい CS 教育カリキュラム J97 の設計方針を中心に述べる。

3.1 J97 の対象

2.2.1 項で述べたように，CS は実証科学 (科学と工学との融合) であり，情報 (コンピュータ) 科学と情報 (コンピュータ) 工学の両方の側面を備えている。従って，J97 の適用対象は，理工系学部の情報系学科である。

具体的には，次の諸点を想定して，J97 は設計されている。

- 大学改革による一般教育科目の再編や大学院重点化に伴う専門教育範囲の時間的かつ空間的広がりにも配慮する。
- 「学部教育と連携あるいは連続した大学院 (特に，博士前期課程) 教育」の必要性が高まっている社会的な背景にも配慮して，大学学部だけではなく大学院 (博士前期課程) のカリキュラムも視野に入れ言及する。
- 「理工系情報学科協議会」の会員校は代表的な J97 の対象大学と想定する。
- 前記協議会の会員校にも，文科系情報学科との大学科，純粋 CS 学科，理工系関連 (電気/電子，数理，機械など) 学科との大学科，大学院大学など多種多様あるが，J97 では，そのような教育組織の枠組みにとらわれず，“CS” という学問分野の教育を対象とする。

3.2 J97 の設計方針

3.2.1 J90 との比較の観点から

J90 では「CS とはこれだ！」を示す必要があったので，その提言では「CS のコアカリキュラム」を中心として策定されている^[1]。しかし，その後の CS の学問的発展/多様化及び CS 教育環境の充実があり，J97 では，その点を踏まえた特徴を出す必要がある。

このような CS の多様化や拡大/拡充化を考えると，J90 のようなコアや標準 (standard) カリキュラムだけではカバーできない分野が生じ不十分である。また，コアカリキュラムでは，3.1 節で述べたような教育組織の多様性 (例えば，大学

科制など)に柔軟に対処できない。

従って、J97では、「コア」や「必修」科目の指定は行わずに、「CSでカバーすべき教育科目をメニューとして列挙する」ことにした。各教育機関の実現可能性(フィージビリティ; feasibility)に合わせてメニューから教育科目を選び組み合わせ、各機関のCS教育カリキュラムを編成してもらう方式である。この方針は、CS専門教育の必要条件の提示を放棄しているが、CS分野の拡大(2.2.5項参照)、初中等教育や社会/産業界との整合性(2.1.2及び2.1.3項参照)などの諸条件を考えると、最適な提示方法である。J97の実現可能性と提示方法との関連については、4.1項でも述べる。

3.2.2 J97の目標

学科における専門教育の目標は、「その学科を卒業したと胸を張れるものを教授してやりたい」であろう。すなわち、この目標をCSに当てはめると、「CS分野外の人が『CSを学んだ人』として見分けられる学生を育てる」こととして良い。

この点を踏まえて、2.2.2項で述べたCSの教育目標に基づいて、J97の目標を次のように設定した。

- 「現実をモデル化し、新しい仕事とする能力」を修得させるとともに、一方で、コンピュータ応用において、モデルと現実との結びつきや関わり合いについて考察できる素養を持たせる。
- 教育課程として、(1)何が問題なのか、どういう方式があるのか?(What?, 問題の定義能力);(2)仕組みはどうなっているのか、どうすれば実装できるのか?(How?, 問題処理機能の実現);(3)どれが良くてどれが悪いのか?(問題処理方法の評価あるいは評価尺度の作り方やモデル化);(4)どうすれば良くなるか、どうすれば良い物を作るか?(より良い実現方法の探求、最適化);という流れが必須であり、J97の設計では、各講義科目ごとにそれに十分配慮する。

後者を一般的に言えば「解けない問題は何か?」、「何が解ければうれしいか?」、「どう変わるとうれしいか?」、「どんな風に役立つのか?」を示すことであり、それがその教育科目の意義や価値を示すために必要である。これに従うCSの専門教

育課程は、2.2.2項で述べたように、「計算機構の直接的実現」あるいは「モデル化した計算機構の実装・評価」であり、これは「モデル化 実装/実現 評価/最適化の繰り返し」とも言える。そのためには、各科目の学習にあたっての前提知識(入力)、およびその科目がどういうことに役立つか(出力)、の両者を示すことが必要であり、J97では、各教育科目ごとにそれらを先修科目、後修科目、関連科目の3項目として明示することにした。(次の3.2.3項の(A)参照)

3.2.3 J97の特徴

3.2.2項で述べた目標のほかに、J97の設計に当たって具体的に考慮し、ACM'91^[7]などと比べて特徴となる事項について述べる。

(A) 教育科目間に関連する項目の明示

CSそれぞれの分野(教育科目)にも基礎から応用までの関連項目やキーワードがあるので、情報処理の機能階層の横割りによってそれを分断することは避けたい。しかし、実現可能性((F)参照)を考えると機能階層を縦貫するような科目(例えば、並列処理など)の設定は難しいと考えられるので、キーワードのグループ化や関連付けによってそれを示す。すなわち、ある科目を中心に、その前提知識としての「基礎科目(先修科目)」、何につながるのかという「応用科目(後修科目)」の、それぞれを示すことによって「キーワード(群)の関連付け」を行う。また、並列処理のような、情報処理機能階層を縦断する分野のキーワードは関連する各科目に取り込むことによって対処する。重要なキーワードは種々の科目で重複出現するので、その度合いが高いものほど重要な項目であると考えれば良い。

この方法は、ACM'91^[7]でも「知識単位(knowledge unit)」として採用されている。J97では、「キーワード(群)」がそれにあたるが、i) キーワードについて、関連付けだけではなく、グループ化も行っている; ii) 単なる項目間の関連付け(先修/後修/関連項目の提示、4.1節参照)だけではなく、「講義科目」というキーワード群における項目(キーワード)の重要性に「修得/知識/関連項目」という重みを付けて示す;などに特徴を出している。

(B) 実験と演習による講義の補完

前の 3.2.2 項で述べた 2 つの目標を実現するためには実験と演習は必須である。特に、実験や演習は「合成やシステム化の重要性を体感させる」という意義がある。さらに、実験や演習は講義を補完するものであるので、J97 では、実験や演習が必要な講義科目については、その内容や実施方法などについて具体的に例示することになっている(4.1 節参照)。また、実験や演習の「適度な困難さ」が学生の達成感を生み出すので、段階的に難度が増すようなテーマ設定などによって、「マニュアルのコピーやマニュアル通りの操作で完了する」、逆に「実装や実現が極端に難しい」実験/演習に偏ることを避ける必要がある。

(C) 歴史(技術史)への言及

歴史(技術史)も教える。歴史については、各教育科目に関連する技術史を教授する時間をとるのが良い。単なる、史実の羅列ではなく、「その技術がなぜ今はダメなのか?」「CS が何を変えたのか?」「CS と社会(現実)とはどう関わり合ってきたのか?」などについて示すことが必要である。

(D) 事例の利用

CS 教育では、講義に事例(case)を用いることが有効である。事例の引用は実現可能性を増す効果があり、事例によって理論を構築することも重要である。ただし、教育では、事例を単に示すだけではなく、「何が重要で何が重要でないかを評価するあるいは評価する能力を育てる」ことが必要である。

(E) 教材及び教育環境

教材の開発や教育環境の整備は重要である。開発した教材や教育環境のフリーソフトウェア化や共有を通して、教育の質の競争が生まれる。J97 では、教材や教育のための良い道具、先端的設備を利用した教育方法についても言及している。例えば、1996 年より大規模システム設計教育研究センター(VDEC)が稼働し、CS 教育に必要なあるいは直接対象となる LSI(マイクロプロセッサ、マイクロコンピュータ)が安価に短期間で試作できる環境が整備されている。VDEC の存在は、CS の特にハードウェア関連教育(実験/演習を含む)方法を一新させる効果が期待できるので、これを大いに利用すべきである。

(F) 教育機関の実現可能性への配慮

J97 では、各教育機関における CS 教育の実状

や実現可能性が多様化していることに配慮している。実現可能性とは「教育機関の規模(小学科/大 学科/大学院一体型/独立専攻/大学院大学など)、教育対象、教員組織、教育環境などによって左右されるカリキュラムの適用(採用)あるいは実現の可能性」である。

3.2.4 J97 における講義科目の設定

講義科目ごとに適切な教育方法には、2.2.4 項で述べたように、情報処理の各機能階層に対応する概念を、i) ボトムアップ教育; ii) トップダウン教育; iii) その両者を混在させる教育; など、多様性がある。これは「ある学問領域を教育する際にそれより下位レベルの概念を適切に隠蔽(ブラックボックス化)する」必要性への対処である。このブラックボックス化の多様性に配慮して、J97 におけるカリキュラム例の提示では、最小粒度のキーワードから最大粒度の講義科目レベルのキーワードのグループ化(関連付け)を活用する。講義科目の設定では、実現可能性を考えると、関連キーワードのグループやレベルによってブラックボックス化せざるを得ない。そして、各講義科目では、その分野で「ブラックボックスであっては困る事項やキーワード」を教授/学習すれば良い。J97 で示す講義科目では、このカバー範囲を明示している。

J97 では、4.1 節で示すように、各講義科目ごとに講述すべき項目をキーワードとして整理し、列挙している。この講義科目の設定における J97 の特徴は以下の通りである。

(1) CS とその境界分野(科目)との関係

CS を支える電気、数学、コンピュータリテラシ、あるいは、その周辺分野である情報システム(IS)、ソフトサイエンス(計算科学)などとの関係については、CS の各分野のカリキュラムの先修あるいは後修科目として位置付けるなどによって、それらとの関連付けを明示することになっている。特に、基礎数学や電気については、ブラックボックス化せずに、J97 で必要なキーワードは取り上げ、関連を明確に記す。

(2) CS 基礎としての情報数学系科目及び計算理論(理論 CS)科目の充実

CS 専門教育は「計算理論を基に情報処理学を展開する学問」と位置付けられる。この「計算理

論」(これを J97 では「理論コンピュータサイエンス(理論 CS)」と呼ぶ)に対応する教育科目として、J97 では、情報数学系 (M-1)~(M-6)、論理回路 (U-1)、形式言語とオートマトン (U-2)、データ構造とアルゴリズム (U-3) などの科目として明示する。さらには、それらとの先修/後修/関連科目付けを行う。これによって、2.1.2項で述べたような「初中等教育と大学の専門教育との整合」をも図る。

また、J97 では、4.2.3項に示すように、理論 CS を中心とするモデル履修コースとして「数理情報科学」コースを設定している。

(3) CS の発展に伴う新規科目の設定および科目内容の拡充

CS の教育/研究分野に関する J90 以降の発展を踏まえて、J90 では言及されなかった講義科目を J97 では設定している。J97 で新しくカバーする分野として、デジタル通信 (U-9)、情報ネットワーク (U-12) などがある。また、J90 で設定されていた科目についても、その後の CS の発展に即して、科目内容を全面的に見直し、拡充を図っている。これに該当する科目としては、情報数学系 (M-1)~(M-6)、ソフトウェア工学 (U-13)、ヒューマン-コンピュータ・インタラクション (U-19) などがある。

(4) ソフトウェア教育とハードウェア教育とのバランス

2.2.2項で述べたように、“CS”は“CS & CE”であり、ハードウェアとソフトウェアを組み合わせる情報処理システムを構築する能力が必須となる。従って、CS 専門教育では、ソフトウェア教育だけに偏るのではなく、ハードウェア教育とのバランスが重要となる。J97 では、例えば、論理設計 (U-6)、集積回路工学 (U-15) などのハードウェア教育科目にも詳細に言及している。

(5) 大学院科目への言及

学部教育科目の設定においては、それと関連する大学院教育も視野に入れる必要がある。J97 では、学部教育と連携あるいは連続して大学院において展開すべき講義科目を「特論 (A-1)~(A-7)」として提示している。また、CS でカバーすべき分野であるが、学部教育の後修科目とした方が適当な講義科目は「学部科目と独立した大学院相当科目 (G-1)~(G-10)」として提示している。

(6) 情報処理階層を縦貫する科目への対処

3.2.2 項の (A) で述べた並列処理などの縦貫科目への対処もキーワードのグループ化(関連付け)活用すれば可能になる。すなわち、このような縦貫科目のキーワード(群)は、関連する科目へ分散配置することによって対処する。

(7) 他学部/他学科におけるコンピュータリテラシ教育との区別

CS 専門教育とコンピュータや情報処理のリテラシ教育とは明確に区別する。具体的に、J97 では、CS 専門学科のリテラシ教育科目として、CS 序説 (L-1) とプログラミング入門 (L-2) とを設けている。また、コンピュータアーキテクチャ (U-4) やプログラミング言語論 (U-5) などをリテラシ教育とは区別した CS 専門教育科目として設計している。

(8) CS の萌芽的あるいは今後展開が予想される境界分野への言及

CS の発展とともに、CS の境界分野は拡大の一途にある。この分野のキーワードは、J97 などの CS 専門教育カリキュラムに反映させることは時期尚早である場合でも、特別講義、セミナー、あるいは CS 教育科目の講義中などに積極的に機会を作って紹介されるべきである。

以下の項目は、境界分野の一例である。

- メディア技術の発展：仮想現実感、拡張現実感、感性情報処理、ヒューマンインタフェース。
- 社会システムへの浸透：電子商取引、電子図書館、CAI、情報公開、知的所有権、情報倫理。
- 脳と知能への接近：認知科学、脳科学、神経科学、複雑系、人工生命。
- 新しい学問領域の創成：計算代数学、計算物理学、計算科学(ソフトサイエンス)、ゲノム情報、発見科学。

4 J97 の提示方式

本節では、新しい CS 教育カリキュラム J97 を具体的にどのようにして提示するかについて述べる。

4.1 J97 の提示項目

J97 は、J90 のように、必要条件としてのコア科目を提示するのではなく、CS 分野全体の教育項目を万遍なく整理系統付けし、それを CS 教育

の十分条件あるいはメニューとして提示する．その J97 を参考にし、各教育機関の実現可能性にあったカリキュラムを組み立ててもらおう方式である．

J97 の具体的な提示方法としては、「各講義科目ごとに、次のような項目について詳細に言及する」方法を採用している．

- 講義科目名： 最大粒度のキーワード群の項目名を「講義科目の名称」(付録 II 参照)として提示している．
- 所要時間： その科目を教授するのに必要な時間数である． $((1.5 \sim 2 \text{ 時間} / \text{週} = 1 \text{ 講時}) \times 14 \sim 15 \text{ 週 (講時)}) = \text{半期} = 2 \text{ 単位分} = [1] \text{ 単位時間}$ として示している．
- 目的： その科目の教授者向け設計思想を述べている．「この科目の狙い」、「何をどういう点に注意して教えたらいいか?」、「こういう風に教えてほしい」、「こういう目標にするのが望ましい」などのこの講義科目の設計目標や設計思想を示している．
- 概要： 学習者にも分かるような科目内容や範囲(到達点)を概要として示している．
- 項目の関連付け： キーワードやそれをグループ化した項目(キーワード群)を講義科目における重要度で重み付けし、i) 修得項目：修得させたい項目；ii) 知識項目：「ああ、あんなことを習ったな」と覚えておいてほしい項目；iii) 関連項目：講義で触れておきたい項目；の3つに分類・提示している．CSの境界分野との関連もここで提示する．
- 講義項目： キーワードを最小粒度とし、それらをグループ化してより粗なレベルの項目名を付け、(1～数講時分の)講義項目として列挙している．それらの講義項目をグループ化した最大粒度の項目が「講義科目名」である．各講義項目の教授に必要な時間数(単位は、 $1.5 \sim 2 \text{ 時間} = 1 \text{ 講時} = [1] \text{ 単位時間}$)も明示してある．
- 先修/後修/関連科目：講義科目間での関連付けである．その講義科目を中心として、i) 先修科目：その科目を履修する前に履修しておくのが望ましい科目；ii) 後修科目：その科目を履修した後に続けて履修するのが望ましい科目；iii) 関連科目：先後は付けられないがその科目と関連する科目；のそれぞれを J97 の科

目名で列挙している．先修科目はその科目を履修するための前提知識であり、カリキュラム設計時には特に注意を払う必要がある．一方、後修科目は「その科目の学習が何につながるのか?」を示す参考資料であり、先修科目ほど厳しい指定事項ではない．先修/後修/関連いずれについても、その科目から見た場合として示してある．従って、ある科目 A の先修/後修あるいは関連科目 B が A を後修/先修あるいは関連科目に指定しているとは限らない．

- 実験及び演習： その科目に連動すべき実験及び演習の題材や内容．実験/演習の環境、所要時間、方法などについてのコメントしている．
- 実施方法： その科目の実現あるいは実施方法についての実現可能性を考えた場合のコメントがあれば示している．他の学部科目あるいは大学院科目との関連や科目名称などへのコメントもここで行う．
- 教科書/参考書： 教科書や参考書を簡単なコメント(内容や特徴についての)付きで紹介している．これらは、入手のし易さ、定評、新しさなどにも配慮して選定している．
- 備考(コメント)： 上記の各項目で書けない、あるいは別項目としたいコメントはここに書かれている．

以上の提示項目を各講義科目ごとにシラバス形式で詳述した J97 本体が付録 III である．

4.2 J97 に基づくモデル履修コース

4.2.1 モデル履修コースの設計目標

3.2節で述べたように、J97 は、CS 分野の拡大に対処するために、CS 教育のコア科目を指定するのではなく、その教育項目を系統的に整理したメニューとして提示している．一方で、教育機関の実状や実現可能性(3.2.3項(F)参照)によって、大学の学部4年間の教育課程として教員が開講したり学生が履修できる科目には、種々の制約が課せられる．CS 分野の著しい拡大が教育機関の実現可能性の多様化を促し、各教育機関ごとに特徴のある CS 教育カリキュラムを組むことができる、あるいは逆に、そうせざるを得ない状況にある．

そこで、J97 で提示した講義科目をもとに、CS 分野の教育カリキュラムとしてのモデル履修コースを数例設定し、読者の参考に供する．ここで提

示する履修コースはあくまでも J97 の適用例であり、これだけが CS 分野の教育課程として考えられる履修コースすべてではない。これらは、各教育機関の特色や教育目標、及び実現可能性に合わせて独自の教育カリキュラムを設計する際の参考としてほしい。

なお、ACM'91^[7]では、「カリキュラムの実現 (implementation) 例」をいくつか詳細に設定している。これに対して、J97では「社会や産業界すなわち CS 専門学科の卒業生の進路 (出力) についても具体的にかつ十分に配慮してモデルコースを設定している」点に特徴がある。

モデル履修コースの設計では、次のような事項を明確に示すことにした。

- 目標：その履修コースの設計思想であり、履修コースを設定した狙いについて端的に述べる。
- 概要：履修コースの学習目標、学習項目、あるいは科目設定の方針などの要点について述べる。
- 卒業者の進路：その履修コースを修めた学生が就くことができる代表的な職業や職種の例についてコメントしている。学部卒業生と社会/産業界との整合を図るための参考資料として利用できる。
- 履修科目：その履修コースの教育カリキュラムの中心となる科目を J97 の講義科目 (付録 II 参照) のうちの学部科目 (L, M, U) の 29 科目) から選んでいる。履修科目の選定方針については、次の 4.2.2 項で詳述する。

4.2.2 モデル履修コース科目の選定方針

モデル履修コース科目の選定方針は次の通りである。

- 各履修コースの教育カリキュラムの中心となる科目を J97 の講義科目から選ぶ。選定する講義科目は「必修科目」ではなく「中心科目」という意味である。
- CS リテラシ科目である CS 序説 (L-1) とプログラミング入門 (L-2) は、原則として、いずれの履修コースにも配当する。
- 各履修コースの中心科目として、上記の CS リテラシ科目以外の学部科目 (M-1 ~ M-6, U-1 ~ U-21) から 10 科目程度選ぶ。

- J97 で示されている先修/後修/関連科目にしたがって、推奨される履修順序で列挙する。
- 各科目の配当期については、教育機関の実現可能性の違いによって左右されるので、ここでは明示しない。

これらのモデル履修コースに沿って学部 4 年間の教育カリキュラムを設計するには、まず、各例で示した科目をコースの中心科目にして、それから、J97 に詳細に示されている各科目の「先修/後修/関連科目」や「実験及び演習」の項目を参考にしながら、各教育機関の実現可能性に合わせて関連科目を設定すればよい。

4.2.3 モデル履修コースの例

CS の専門学科用として例示するモデル履修コースは次の通りである。

- [A] 情報機器工学
- [B] コンピュータ工学
- [C] ソフトウェア科学
- [D] ソフトウェア設計工学
- [E] 知能情報学
- [F] マルチメディア情報処理
- [G] 情報ネットワーク学
- [H] 数理情報科学
- [I] 人間情報科学

[A] ~ [I] の各履修コースは理工系学部情報系学科の学部での教育課程例であり、多様な CS 分野でそれぞれ代表的かつ特徴的な履修コース例として示している。これらの 9 つの各履修コースは、ブラックボックス化 (3.2.4 項参照) している知識レベルが異なり、その相違によって特徴付けられている。

また「情報系学科における CS 専門教育としてではない」CS に関する履修例 (モデル) として次の 2 つを示す。

- [X] CS 外学科卒業生が CS 分野へ移行してくる際に必要な CS 最小セット
履修コース [X] は、情報系学科以外の学生が情報系学科に編入する、あるいは大学院で情報系 (CS 分野) を専攻する際に、学部入学時から CS 教育カリキュラムを受けてきた学生と同程度の CS 分野の基礎知識を持つために最小限学習しておくべき科目を示している。
- [Y] CS 外専攻大学院生のための CS 最小セット

表 1: J97 の講義科目とモデル履修コース

科目	A	B	C	D	E	F	G	H	I	X	Y	科目名
L-1												CS 序説
L-2												プログラミング入門
M-1												情報代数と符号理論
M-2												離散数学
M-3												計算論
M-4												確率論と情報理論
M-5												数理計画法
M-6												数理論理学
U-1												論理回路
U-2												形式言語とオートマトン
U-3												データ構造とアルゴリズム
U-4												コンピュータアーキテクチャ
U-5												プログラミング言語論
U-6												論理設計
U-7												オペレーティングシステム
U-8												コンパイラ
U-9												デジタル通信
U-10												データベース
U-11												人工知能
U-12												情報ネットワーク
U-13												ソフトウェア工学
U-14												数値計算
U-15												集積回路工学
U-16												信号処理
U-17												画像情報処理
U-18												パターン認識
U-19												HCI
U-20												コンピュータグラフィックス
U-21												自然言語処理
コース名	情報機器工学	コンピュータ工学	ソフトウェア科学	ソフトウェア設計	知能情報学	マルチメディア	情報ネットワーク	数理情報科学	人間情報科学	CS 分野への移行	CS 外専攻大学院生	

CS分野で教育して獲得できる知識の中には、情報系学科以外の学生にとっても、情報技術 (IT) や情報リテラシとして必須のものがある。これを履修コース [Y] として示している。[Y] は、CS 分野を主専攻としないか CS 分野を副専攻とする理工系大学院生が修めておけば、主専攻分野での研究に役立つ CS 分野の講義科目を示している。

[X] と [Y] の両履修コースについては、[A] ~ [I] の半分の 5 科目程度を中心科目として J97 の講義科目から選んである。

なお、各履修コースには「コース名称」が付けられているが、これは各履修コースの特徴が直感的に識別できるように便宜上付けたものであり、その履修コースの学問分野の総称として周知されていたり使用されている呼称ではない。

付録 IV では、各履修コースの目標、概要、具体的な履修科目 (推奨履修順序) などの各項目について詳細に示している。

4.2.4 J97 の講義科目とモデル履修コース

J97 の講義科目と各モデル履修コースとの関係については、表 1 に示している。

[A] ~ [I] の履修コース例では、[L] の 2 科目以外の [M] と [U] 科目から 10 ~ 13 科目が各履修コースの中心科目として選定されている。

11 種類の履修コース例の半分以上で選定された講義科目は、計算論 ([M-3])、形式言語とオートマトン ([U-2])、データ構造とアルゴリズム ([U-3])、コンピュータアーキテクチャ ([U-4])、プログラミング言語論 ([U-5])、オペレーティングシステム ([U-7])、コンパイラ ([U-8]) の 7 科目である。

5 おわりに

本報告書では、日本の大学における新しい CS 教育カリキュラム J97 の概要について説明した。

実際の J97 は、講義科目の設定、その詳細化によるシラバス例、履修モデルコースの詳細説明を中心にまとめられ、この報告書の付録 III、IV として提示してある。

謝 辞

J97 の設計においては、J97-WG 委員以外の諸先生方にもそれぞれの専門的立場から協力して頂いた。

以下に、その御名前を印し、感謝の意を表す。魚井 宏高 (大阪大学)、河原 達也 (京都大学)、佐藤 宏介 (奈良先端科学技術大学院大学)、名取 亮 (筑波大学)、横田 一正 (岡山県立大学) の諸先生方である。

また、J97-WG では、情報処理学会・情報処理教育カリキュラム調査委員会における議論を大いに参考にしている。この委員会のメンバー諸氏に深謝の意を表す。

なお、本研究の一部は文部省科学研究費補助金・基盤研究 (B)(No.07308025) 及び文部省委嘱調査研究費による。

参考文献

- [1] 大学等における情報処理教育検討委員会: 大学等における情報処理教育のための調査研究報告書, 情報処理学会 (1991 年 3 月).
- [2] 野口 他: 大学等における情報系専門教育の改善への提言, 情報処理, 情報処理学会, Vol.32, No.10, pp.1079-1092 (1991 年 10 月).
- [3] 牛島 和夫: 理工系情報専門学科におけるコアカリキュラムについて, 情報処理, 情報処理学会, Vol.32, No.10, pp.1093-1100 (1991 年 10 月).
- [4] 都倉 信樹: 情報処理教育における実験・演習, 情報処理, Vol.32, No.10, pp.1101-1108 (1991 年 10 月).
- [5] Denning,P.J. et al.(木村 泉:訳): 学問としての計算機分野, 情報処理, 情報処理学会, Vol.31, No.10, pp.1351-1372 (1990 年 10 月).
- [6] 國井 利泰(編・監訳): コンピュータサイエンスのカリキュラム, bit 別冊, 共立出版 (1993 年 1 月).
- [7] Tucker,A.B. et al.: Computing Curricula 1991 — Report of the ACM/IEEE-CS Joint Curriculum Task Force, acm PRESS (1991).
- [8] UNESCO and IFIP: A Modular Curriculum in Computer Science (1994).
- [9] 中森 真理雄: 情報工学の次世代カリキュラムに関する諸条件 - 大学改革の中で -, 「新しい時代の情報処理教育カリキュラム」シンポジウム・論文集, 情報処理学会, pp.71-78 (1994 年 10 月).
- [10] 日本電子機械工業会: 半導体産業『創造的研究開発への挑戦』, pp.1-38 (1995 年).
- [11] 都倉 信樹: 小中高一貫情報カリキュラムへの期待, 「小中高一貫情報教育カリキュラムの構成と総合化の可能性」シンポジウム・論文集, 日本教育工学会, pp.29-34 (1996 年 6 月).
- [12] 武井, 大岩: 高等教育との接続性からみた情報教育, 情報処理, 情報処理学会, Vol.38, No.9, pp.811-819 (1997 年 9 月).
- [13] 中村, 松田: 情報教育担当教員の養成, 情報処理, 情報処理学会, Vol.38, No.10, pp.923-928 (1997 年 10 月).

付録 I : J97-WG 委員名簿

- 柴山 潔 (京都工芸繊維大学 [主査])
- 横矢 直和 (奈良先端科学技術大学院大学 [副査])
- 荒木 啓二郎 (九州大学)
- 石田 亨 (京都大学)
- 竹村 治雄 (奈良先端科学技術大学院大学)
- 都倉 信樹 (大阪大学)
- 並木 美太郎 (東京農工大学)
- 萩原 兼一 (大阪大学)
- 福田 晃 (奈良先端科学技術大学院大学)
- 松本 裕治 (奈良先端科学技術大学院大学)
- 美濃 導彦 (京都大学)
- 村田 正幸 (大阪大学)
- 安浦 寛人 (九州大学)
- 渡辺 治 (東京工業大学)

付録 II : J97 の講義科目

L CS リテラシ科目

- L-1 コンピュータサイエンス序説
- L-2 プログラミング入門

M 学部科目 (情報数学系)

- M-1 情報代数と符号理論
- M-2 離散数学
- M-3 計算論
- M-4 確率論と情報理論
- M-5 数理計画法
- M-6 数理論理学

U 学部科目

- U-1 論理回路
- U-2 形式言語とオートマトン
- U-3 データ構造とアルゴリズム
- U-4 コンピュータアーキテクチャ
- U-5 プログラミング言語論
- U-6 論理設計
- U-7 オペレーティングシステム
- U-8 コンパイラ
- U-9 デジタル通信
- U-10 データベース

- U-11 人工知能
- U-12 情報ネットワーク
- U-13 ソフトウェア工学
- U-14 数値計算
- U-15 集積回路工学
- U-16 信号処理
- U-17 画像情報処理
- U-18 パターン認識
- U-19 ヒューマン・コンピュータ・インタラク
ション (HCI)
- U-20 コンピュータグラフィックス
- U-21 自然言語処理

A 大学院科目 (学部科目の特論)

- A-1 論理回路特論
- A-2 コンピュータアーキテクチャ特論
- A-3 オペレーティングシステム特論
- A-4 データベース特論
- A-5 画像情報処理特論
- A-6 コンピュータグラフィックス特論
- A-7 信号処理特論
- A-8 ソフトウェア工学特論

G 大学院科目

- G-1 ハードウェアアルゴリズム論
- G-2 論理設計と CAD
- G-3 プログラム意味論
- G-4 ネットワークアーキテクチャ
- G-5 分散処理システム
- G-6 情報セキュリティ
- G-7 関数プログラミング
- G-8 オブジェクト指向プログラミング
- G-9 音声情報処理
- G-10 アルゴリズム論

付録III：J97の講義科目（詳細）

(凡例)

識別番号 講義科目名 [所要時間]

最大粒度のキーワード群の項目名を講義科目名 (付録 II 参照) として提示している .

また, 所要時間は, その科目を教授するのに必要な時間数である . ((1.5~2 時間 / 週 = 1 講時) × 14 ~ 15 週 (講時) = 半期 = 2 単位分 = [1] 単位時間) として示している .

目的

その科目の教授者向け設計思想を述べている .

概要

学習者にも分かるような科目内容や範囲 (到達点) を概要として示している .

項目の関連付け

キーワードやそれをグループ化した項目 (キーワード群) を講義科目における重要度で重み付けし, i) 修得項目; ii) 知識項目; iii) 関連項目; の 3 つに分類・提示している . CS の境界分野との関連もここで提示する .

[修得項目]

修得させたい項目

[知識項目]

「ああ, あんなことを習ったな」と覚えておいてほしい項目

[関連項目]

講義で触れておきたい項目

講義項目

キーワードを最小粒度とし, それらをグループ化してより粗なレベルの項目名を付け, (1~数講時分の) 講義項目として列挙している . それらの講義項目をグループ化した最大粒度の項目が「講義科目名」である . 各講義項目の教授に必要な時間数 (単位は, 1.5~2 時間 = 1 講時 = [1] 単位時間) も明示してある .

先修/後修/関連科目

講義科目間での関連付けである . その講義科目を中心として, i) 先修科目; ii) 後修科目; iii) 関連科目; のそれぞれを J97 の科目名で列挙している . 先修/後修/関連いずれについても, その科目から見た場合として示してある . 従って, ある科目 A の先修/後修あるいは関連科目 B が A を後修/先修あるいは関連科目に指定しているとは限らない .

先修科目 その科目を履修する前に履修しておくのが望ましい科目である . 先修科目はその科目を履修するための前提知識であり, カリキュラム設計時には特に注意を払う必要がある .

後修科目 その科目を履修した後に続けて履修するのが望ましい科目である。後修科目は「その科目の学習が何につながるのか？」を示す参考資料であり、先修科目ほど厳しい指定事項ではない。

関連科目 先後は付けられないがその科目と関連する科目である。

実験及び演習

その科目に連動すべき実験及び演習の題材や内容、実験/演習の環境、所要時間、方法などについてのコメントしている。

実施方法

その科目の実現あるいは実施方法についての実現可能性を考えた場合のコメントがあれば示している。他の学部科目あるいは大学院科目との関連や科目名称などへのコメントもここで行う。

教科書/参考書

教科書や参考書を簡単なコメント（内容や特徴についての）付きで紹介している。これらは、入手のし易さ、定評、新しさなどにも配慮して選定している。

備考（コメント）

上記の各項目で書けない、あるいは別項目としたいコメントはここに書かれている。

CS リテラシ科目 **L**

コンピュータサイエンス序説 **L-1**

プログラミング入門 **L-2**

L-1 コンピュータサイエンス序説 [1]

目標

この科目では、

- A．日常の大学生活でコンピュータを活用する基本的能力の養成
- B．コンピュータサイエンス分野への導入
- C．コンピュータに関する基礎的知識
- D．大学生活とその後の進路についての案内と学習への動機付け

などを目的とする．Aは狭義のコンピュータリテラシを扱う．Bはコンピュータサイエンスの学問分野の概観を与え、これからの学習の方向を指し示す．高校ではコンピュータ教育をあまり受けていないので、基礎的な知識を全く持たない入学生が実際には多い．個人的にパソコンをかなり使う学生もいるが、系統だった基礎知識を必ずしももたないので以後の学習に必要な最小限の基礎的な事柄を系統だって学習する入門的内容がCである．Dは大衆化されたといわれる昨今の大学で学生に大学生活のあり方や意義、ノウハウ、して良いことと悪いこと、大学という社会での約束事などを伝える．高校の延長のままの考えを脱却できない学生もときとしてあるが、大学での学習についてよく指導し学習の案内と動機付けをはかる．このようなやや雑多な内容であるが、これは各大学・学科の事情で、濃淡をつけてよい．以下、A～Dの内容に相当する事項をあげるが、その取捨選択をしていただければよい．

A．コンピュータリテラシ

1. キーボード操作

特に、タッチメソッドを早期に完全に習得するように指導することが望まれる．それによって、入力効率は大幅に向上し、日常的にコンピュータを違和感なく使うようになる．

2. フロッピーディスクなどの扱い方

3. 日本語入力の方法

4. メールの送受信

メールはできるだけ早期に使えるように指導するとよい．これによって、コンピュータは通信の手段になる．教官からメールを送ったり、学生の自己紹介などをメールで提出させて、日常的な通信手段となるようにする．

5. 日本語ワードプロセッサ、プリンタの使い方

レポートはワードプロセッサで作成することを要求し、習熟させる．

ほぼここまでが、最初の必須事項であろう．以後、次のような応用ソフトの存在を知らせ、必要なものから使用経験を積ませ、応用プログラムを活用するという方法を知らせる．

6. WWW、ニュースなどとネットワークリテラシ

7. ドローツール

レポートの説明図などから使用経験を持たせる．

8. 表計算やマセマチカなどの数式処理プログラム

これも実験や演習の必要から使用する形で自然に導入できればよい．

9. データベースソフト、等々

B．コンピュータサイエンス分野への導入

1. 学問分野としてのコンピュータサイエンスの概観
2. 学科のカリキュラムの編成・前後関係の図などを示し、どの辺までをどの程度学習するかを説明する。
3. 各グループ（講座）からの 20 分か 30 分くらいの短い研究内容の紹介をする。そういう研究をするのに、必要な予備知識となる科目の説明をすることも、特に低学年配当の科目の受講動機を与える意味で必要であろう。
4. コンピュータの活用されている事例の紹介などを行い、どういうところに研究成果が生きているかなどを説明する。つまり、この分野の学習の有用性を示す。

C．コンピュータに関する基礎的知識

コンピュータ概論などの名称で行われている講義の内容である。幅優先のアプローチで概観を与える。

1. コンピュータとはなにか？
2. どういうところで使われているか？
3. 中はどうなっているか？
4. ハードウェアの基礎知識
5. ソフトウェアの基礎知識
6. 通信の基礎知識
7. コンピュータの応用
8. コンピュータの可能性と限界

D．大学生活とその後の進路についての案内と学習への動機付け

1. 高校と大学とのシステムの違い

大学の文化を知らせ、大学生としての生活の仕方を伝える。大学での慣習や約束ごとなどを伝えて、大学というコミュニティーの一員として早くとけこむことを可能とする。これによって、大学での学習と高校までの学習の違いをわからせ、大学での学習の方法を伝える。

図書館、インターネットによる情報取得の方法
学会の紹介

2. 学科・コースの特徴やカリキュラムの構成、学習計画の立て方

どういう目標を立てるべきか？
タイムスケジュール
大学院との接続、進路等の話

3. 専門家、職業人を目ざす心構え、技術や科学についての態度・考え方

職業人・専門家としてのプライドと責任感を持たせる最初のステップ（これは以後おりに触れて、どの科目でもその意識を高める配慮が望まれる。ここではその最初のステップである。）
抽象論でなく、過去の事例を通して、現在でのその事例の意味付け、評価を与えることで、歴史に学ぶ科学・技術哲学的アプローチが効果的であろう。

この科目については、以上のように種々の要素が考えられる。したがって、ひとつのシラバスを提示する事はほぼ意味がないので、実施方法や補足の欄で実現事例や案を示すことにしたい。概要も科目内容で変わるので省略する。ただ、新入生を対象とする場合は、高校生に理解できる言葉を選んで書くようにしたい。

補足説明

専門学科の場合この種の科目が必要かどうかは議論のあるところであろう。J90には少なくともこのような科目のアイデアは見られない。なぜJ97でこの科目を登場させるのか？その理由をあげてみよう。

1. いかにか専門学科といえ、突然プログラミングやコンピュータの原理などの科目で詳しい話をするのが効果的かという点必ずしもそうとはいえない。特に、最近はプログラムしなくても多くの有用なソフトウェアが利用でき、まずそれらをいろいろ経験することから始めた方が、プログラミングやコンピュータの原理等の科目への動機付けが図れる。
非専門学科の学生が種々のソフトを使っているのに、専門学科の学生はそういうソフトに触る余裕もなく、特定の言語でのプログラミングに苦闘するという方法はかならずしも有用とは思われない。
2. それらの種々のソフトウェアを使うことから始めて、なぜこういうことができるのだろうかという誰しも思う疑問を引出し、興味を引き付けてから、その仕組みや原理を伝えるという方法は、動機を与えるという意味で効果的である。また、そのようなすばらしい働きをするプログラムも結局は人間が作っている、それをプログラミングという科目で学ぶという意味付けをすれば、学生も自然に動機づけられるであろう。
3. 有用なソフトウェアの存在も知らず、まして、使ったこともないということでは、さらに優れたソフトウェアや計算機システムを創り出そうということにも結び付かない。優れたソフトウェアをできるだけ経験することが、よりよいソフトウェアを創り出すことに結び付くであろう。
4. 情報学科はプログラムを書くことを学ぶところであるという誤解をしている学生も少なくないが、そうでなくとも最初にプログラミングを教えるとそれがすべてであるという錯覚をもち、その他の重要なことに目がいかないようなことにもなりかねない。まず、この科目で、ワードプロセッサや表計算、メール等の種々の道具の使い方を学び、レポートなどもこれらの道具を用いるようにさせ、プログラミング科目でも単にプログラムを書くだけでなく、仕様や設計、使用法の説明等の多くの文書を書く力を付けることを考えるべきであろう。また、プレゼンソフトを用いての説明の練習なども有用であろう。
5. また、新入生にオリエンテーションコースを履修させることを推奨する意見もある。

[Coo96] C.R.Cook:A Computer Science Freshman Orientation Course, SIGCSE Bulletin, Vol.28, No.2, pp.49-55,61, June 1996.

この論文のアブストラクトを引用しよう。

This paper describes an orientation course for beginning computer science majors. The course is different from the CS1 or computer literacy courses, but similar in intent and content to orientation courses in engineering, business, and other fields. Its purpose is to give students an overview of computer science and an idea of what computer professionals do so that students can make an informed career decision.

Other emphasis for the course are practice in problem solving, experinece working in groups, teaching basic technical (non-programming) skills, social and ethical issues, and making students aware of the resources and opportunities availble to them such as internship programs. Influences and constraints on the design of the course and suggested changes the next time the course is taught are also described.

こういう考え方もあるということである。実際にこの著者の設計が日本で利用できるかは即断できないが、米国でもいろいろの発想を出し、検討をしているということである。

以上のような存在理由をあげることができるが、もう少し違う見方も指摘しておきたい。

リテラシ教育というと、いくつかのツールの使用法の経験と、その際のマナーなり注意事項（メールや WWW を使う際のしつけなど）、コンピュータを扱う上での種々のノウハウ（フロッピーディスクの初期化、パスワードの付け方など）、タッチタイピングや疲労や障害の起こらない使い方などの注意など広範囲のことがらも実際には含まれるであろう。

ただ、専門学科の場合、このような非専門学科と同じような内容でいいのかという疑問もあろう。そういう疑問が生じるのは、単なるユーザーでなく、将来それらのツールやシステムを作ったり、計画する立場になるということから、単純ユーザーとは当然なにか違う取組みがあるのではないかという問題の指摘であろう。

この答えは、取り上げる素材やツールは仮に同じであっても、つねに単なるユーザーと違う着目点を指示したり、誘導することは可能であり、それが以後の学習への有効な動機付けになることを意識しての指導をするということである。

そのような例は随所にあると思われる。少し挙げてみる。

- マウス

どういう仕掛けか？

どのようにコンピュータに信号が伝えられているのか？

マウスを盛んに動かすと他の表示が遅くなるがなぜか？

- メール

字化けのする例をとらえて、字化けの理由は？

一体こういう字をコンピュータではどう表現しているのか？

なぜメールが伝わるのか？

コンピュータからどんな線につながっているのか？

- ワードプロソフト

ローマ字かな変換はどうしているのだろうか？

かな漢字変換はどうしているのだろうか？

そもそもどうして、こんなに漢字を扱えるのか？

使っている内に漢字の変換の仕方がなじんでくるのはなぜか？

どんどん疑問を持たせ、一部は適切に回答し、一部は3年の という講義で本格的に説明するから楽しみにしているようになどと、後に続く講義への関連付けをしていく。非専門学科はその場限りに近いが、専門学科はこれらの疑問に直接間接に答える科目が後に続く点が違う点であり、そのことを意識して指導することで、この科目の意義は大きいものとなる。

この科目の内容は、各学科のカリキュラム構成や、使用できる機材、ソフトウェアによって大きく影響されるので、さきに挙げたような内容をすべてやらなければならないという意味ではない。むしろ、この科目に含まれ得る候補を挙げてあるという意味である。

上で一応構成要素案を示したが、これはあくまで一案に過ぎない。各大学の実状に応じて適切にアレンジすることを期待する。この議論ではもう少し違った観点から、この科目の別のあり方を議論してみたい。

リテラシ科目をおく理由として挙げた要因は、つまるところ、まず最初にプログラミングとかコンピュータの構成や原理を教えようとするのでなく、一種の緩衝材としての役割を期待しているといえるであろう。本題にすぐとりかかるより、いったん間をおくことで、より本題への動機付けを行い、効果的学習を期待するという考え方である。それならば、別にアプリケーションの使用法をやるのでなくともよいのではないかという意見もありえよう。その考え方の一つは上の CookCoo96] の論文にある問題解

決についてであろう。また、グループワークとプレゼンテーション，日本語での文書作成なども候補にあげられる。

実現上の留意事項

上記は基本的な考え方であり，一律にどこまでを教えないといけないということではない。いくつかの内容を適当に混ぜ合わせるという立場であり，無計画に進めると散漫な講義となり，効果が上がらないであろう。講義計画を十分練り，シラバスを作成し，実際のコンピュータを使っての演習と，座学を利用可能な学習環境の範囲で適切に配分する必要に迫られよう。特に，コンピュータをはじめてさわる最初の時期では，できるだけ TA や教員の応援を増やしてきめ細かい指導をするということが望まれる。また，学科の先生の協力を得て，B.3(16ページ)にのべた自分の学科での研究の特色を学生に示し，期待を持たせることも有用であろう。そういう意味で複数の教官の担当協力が必要で，先にも述べたように，綿密な計画が必要となる。

毎回簡単な課題を与えて，電子メールでレポートさせるなどの方法で，単に一度操作したというだけでなく，習ったことがすぐ役立つという感じにできるとよい。実は，上にあげた項目は2単位ですませることは無理で，当然取舍選択が必要となる。後続の科目との関係でどこまでをカバーするかを決める必要がある。

実施方法

ここではいくつかの例とか考え方を紹介する。

1. いわゆる幅優先方式の CS (コンピュータサイエンス) 入門

数年前から，米国では breadth first approach という考え方が提唱され，いくつかの教科書が書かれ，それをういた実践もいくつかの大学で行われている。その基本的な考えは，それまでの縦一列に積み上げていくカリキュラムを反省し，まず初学者に CS という学問分野の概観を与え，CS の魅力を伝え，以後の学習に動機付けしようというものである。また，こういうことを言われた先生もある。アメリカでは，コース間の流動が自由であるため，CS1 はとるが，その後は CS の講義は聴かなくなる学生もかなりある。CS への登録学生数が伸び悩み始めたということもあり，なんとか CS につなぎ止めたいという意図もあるという。また，仮に CS1 しか聴かないとしても，それだけで，CS 全体と判断されるのは残念で，できるなら CS についての概観を得てほしい。それが，他の分野の人にも CS という学問分野についての理解を深めるのに有用であるという考えでもあるという。この考えで書かれた教科書(教科書/参考書の項を参照)をあげるが，それでこのアイデアをみていただきたい。日本の場合，従来は自分の学科に来た学生はまず他へうつることはなかったから，こういう発想は必要なかったかも知れない。しかし，大学科に移行する大学も増えており，その場合多くの学生のうちからある割合の学生が自分の分野に配属されることになる。当然，自分の分野の魅力を学生に伝える科目の必要が生じる。実際，大学科になったところでは，情報工学序論とか情報工学入門などの科目を設定し，学生に内容の紹介をする例が増えている。そこでは，時間が必ずしも十分とれないため，米国の教科書をそのまま講義することは難しいが，その考え方は参考にはなるう。

2. 大学科での分野紹介的科目

たとえば，京都大学情報学科では，1 研究室 4 コマを使い，自分の研究分野のサーベイとか基本概念などを説明するという講義を開講している。通年講義。各先生がもっとも得意とするところを熱意を込めて話をされるようで，ある意味で一番面白い話の聞ける講義であるということである。

3. 琉球大学の修学計画 I, II

琉球大学では修学計画 I, II という非常にユニークな講義を行っているということを知り、そのシラバスをいただいた。これは最初にあげた B, D などの実際の指導の例であるとともに、グループ活動などを含め、単に座学でない参加型の科目としている。学生に調査討論をさせ、自分のキャリアパスを考えさせているのも参考になる。なお、ゲスト講師のお名前等はここでは A 氏などとさせていただきます。

-
1. 科目名：情 109 修学計画 I 必修 1 単位 (1-0) 情報工学科 1 年生 前期
毎週 木曜日 4 限目 (14:40-16:20) 院 111 教室
 2. 担当教官：翁長健治 教官室：電情 408
 3. 授業内容：入学早々の第 1 学期に於いて、大学教育の理念と組織/形態を良く理解し、自らの修学計画に良い見透しを得ることを目的とする。この目的を達成する為に
 - 1) 大学の仕組み、学部の役割、学科の陣容、教官の研究教育等の解説
 - 2) 学科カリキュラムの構造と人材育成の狙い、卒業生の活動状況
 - 3) 産業界の将来動向とキャリアパスの形成等について、1 グループ 6 名、10 程度のグループに分かれて集団討議/調査/報告を通じて学習を行う。
 4. 履修条件：特になし。
 5. 教科書：各種資料を随時配布する。
 6. 評価方法：
 - 1) グループ調査報告書の提出
 - 2) ゲスト講師の講義に対するレポートの提出
 - 3) 最終評価は上記 2 者を総合的に判定して決める
 7. ゲスト講師：
A 氏 (沖縄県物産公社)
B 氏 (横河電機)
C 氏 (沖縄県庁情報システム課)
D 氏 (トロピカルテクノセンター情報開発部)
 8. ゲスト講話者：未定
 9. 授業計画：

週	内容
1 (4/13)	グループ分け、学習方針の説明
2 (4/20)	グループ調査テーマの割り当て：沖縄県概況調査 (行政全般、農業、水産業、製造業、サービス産業、新聞、印刷業、交通運輸業、観光産業)、第三次沖縄振興計画概況調査、琉球大学概況調査、A4、10-20 ページ程度の報告書、調査日程表、作業分担表
3 (4/27)	学科陣容の紹介

- 4 (5/11) 大学の仕組み，学部の役割
- 5 (5/18) ゲスト講義 I：「産業社会論」A氏
- 6 (6/01) ゲスト講義 II：「企業と人」A氏
- 7 (6/08) ゲスト講義 III：「人と仕事」A氏
- 8 (6/15) ゲスト講義 IV：「行政に於けるコンピュータ」C氏
- 9 (6/22) ゲスト講義 V：「情報産業における技術革新」D氏
- 10 (6/29) キャリアパスの形成についての討論 I
- 11 (7/06) キャリアパスの形成についての討論 II
- 12 (8/30) 夏休み合宿：グループ調査の報告
(8/31) ゲスト講話：「青春時代の人生観」講話者未定
- 13 (9/07) ゲスト講義 VI：「仕事のできる技術者とは」B氏
- 14 (9/14) 教官の研究紹介 I
- 15 (9/21) 教官の研究紹介 II

- 1. 情 209 修学計画 II 必修 1 単位 (1-0) 2 年前期
- 2. 担当教官：高良富夫 (情 404 室)
- 3. 授業内容：情報工学科で 1 年間学んだことをふまえ，今後の修学の計画をする．情報工学科の学生が将来，職場として選択するであろうところで，現在活躍しておられる方をゲスト講師として迎え，その職種の社会的位置付け，現状と将来，毎日の業務などについて語っていただき，自分の進路について理解を深め考究する．自分の進路を想定して，卒業までの修学計画をする．
- 4. 履修条件：修学計画 I を履修していること．
- 5. 教科書：「修学手引書」，情報工学科，平成 7 年度．また，各種資料を随時配布する．
- 6. 評価方法：出席状況，報告書の結果を総合して判定する．
- 7. 授業計画
 - 4/12 (1) 前期開講科目の紹介
 - 19 (2) 前期開講科目の登録
 - 26 (3) グループ研究課題の検討
 - 5/10 (4) 講話「研究公務員」：A氏 (情報工学科教授)
 - 17 (5) 講話「企業における研究」：B氏 (情報工学科助教授)
 - 24 (6) グループ討論
 - 31 (7) 講話「工業系教員」：C氏 (元工業高校校長)
 - 6/07 (8) 講話「SEの仕事」：D氏 (IBM 沖縄営業部)
 - 14 (9) グループ討論
 - 21 (10) 講話「ソフトウェアの研究開発」：E氏 (沖縄日電)
 - 28 (11) 講話「電力会社における情報部門」：F氏 (沖縄電力)
 - 7/05 (12) 合宿の準備
 - ? (13) 企業見学，合宿，調査結果発表会
 - 9/13 (14) グループ討論
 - 20 (15) 最終レポート提出

以上，3 つの例を示した．

教科書/参考書

このような構想にぴったりの教科書は見あたらない。部分的にカバーするものをいくつかあげる。昨今の学生は教科書も薄いものでないと嫌うというような風潮もある。読書力を付ける意味でも、適当な副読本がえられればいいが、これもあまり適当なものがないのが実状であろう。よい教科書、副読本の出版が待たれる。手元にあるものをあげるので、網羅的ではないし、よい本が漏れていることをおそれるが、参考までにリストしておく。

1. P. グロゴノ, S. H. ネルソン著, 永田守男 訳: 問題解決とプログラミング, 近代科学社, 1985, ISBN4-7649-0104-8 C3350 ¥2400

問題解決という視点から入っており, 1年生にも適している。

2. A. B. Tucker, W. J. Bradley, R. D. Cupper and R. G. Epstein:

- Fundamentals of Computing I, Logic, Problem solving, Programs, and Computers,
- Fundamentals of Computing II, Abstraction, Data structures, and Large Software Systems,
- Fundamentals of Computing I, Pascal Laboratory Manual,
- Fundamentals of Computing II, Pascal Laboratory Manual,

McGraw-Hill

いわゆる breadth-first approach の教科書で, 明確な方針に基づいている。

3. A. Biermann: 和田 監訳: やさしいコンピュータ科学, アスキー

これも breadth-first approach の教科書として使われている。

4. J. Impagliazzo and P. Nagi: Computer Science A Breadth-First Approach with C John-Wiley&Sons,Inc. ISBN0-471-58552-1.

タイトルに breadth-first approach と示されている。

5. 松浦敏雄, 中西通雄, 原田章 監修: NEXTSTEP によるコンピュータ・リテラシー入門, ASCII, 1996, ISBN4-7561-0791-5

大阪大学情報処理教育センターの作成した通常の意味のリテラシ教科書

6. 川合慧: 「コンピューティング科学」東京大学出版会, 1995, ISBN4-13-062138-6

東京大学教養学部での総合科目「計算機科学」の講義の3年間の経験から生まれた本。コンパクトな本であるが, 広い題材をカバーしている。

7. 遠藤諭: 計算機屋かく戦えり, アスキー, 1996, ISBN4-7561-0607-2

副読本に使えるようなものを見本としてあげておく。

備考 (コメント)

ここでは, 備考として, 倫理教育, 歴史教育等にふれる。

1. 初等中等教育でのリテラシ教育では, いわゆる倫理教育が含まれているが, それはどうするのか? もちろん, 大学でもその内容を入れることは考えられる。現況を考慮すると, それは講義の進行に沿って, 要所要所でふれるのが適当であろう。仮に教科書に, コンピュータ社会の倫理などという単元があって, それをそのまま説明する知識伝達型の講義でやってもあまり意味はないかもしれな

い。ただし「情報倫理」という2単位の科目が全く成立しないというつもりはない。しかし、いくつもの問題がある。いつやるのか。鉄は熱いうちに打てというが、1年生が適当か？ 1年生にそういう話を2単位分やっても興味をそくだけかもしれない。では、3年生か？ 多くの大学は3年生に専門の科目を多数配しており、この種の科目は割込みにくいかも知れない。比較的可能性があるとすれば、4年生かもしれない。米国ではすでに数種類の教科書があるが、日本ではどうだろうか？ また、これを自信をもって担当できる教員はおられるかも知れないが、少数であろう。つまり、近い将来を考えても、独立科目と出来るほどの状況にはない大学が多いのではないだろうか。しかし、専門分野についての職業倫理や社会との関わりについて、さけて通ることはもはや許されない。現状で、可能と思われることを少し述べておく。

最近の入門の教科書は、ethical and social issue 等の章を設けているので、これを参考に、たとえば、この科目の中で数回の時間をとることは専門分野への第一歩を印す学生にとっても有用であろう。ただ、一回や二回倫理教育をしましたとあって、あとほったらかしにするということではなく、最初にどうしても言うべきこと（例：パスワードの管理）は言うが、メールも使えないうちから著作権などといってもあまり実感がわかない。むしろ、倫理教育はこの科目で着手するにしても、この科目だけの責任ではなく、後続の実験演習を含め、つねに適切なタイミングで伝えていくのが望ましいと考える。また、一人の教員なり学生が違反行為をしてみせたら、この教育は無に帰する危険性がある。学科のメンバーが同じ意識にたって行動し、折にふれて学生に伝えるということが望まれる。

なお、学会の倫理綱領（IEEE, ACM, 情報処理学会等で制定されている）を学生に示し、その基本的な考え方を伝えるというのは非常に効果的と思われる。単に、あれをしてはいけない、これをしてはいけないという話でなく、専門家として立つ意識から出発することが、より効果的ではないだろうか。高度の専門知識と技能を有する専門家としてのよい意味のプライドをもたせることが学生の支えになるのではないだろうか？

実際には、倫理教育は言うのは簡単ではあるが、実施はなかなか困難を伴うと予想される。これについては、たとえば、次の論文が参考になる。

K.G.Schulze and F.S.Grodzinsky: TEACHING ETHICAL ISSUES IN COMPUTER SCIENCE: WHAT WORKED AND WHAT DIDN'T Proc.ACM SIGCSE'96. pp.98-101, 1996.2.

この論文はいろいろの教育法についてどういう方法があるか、うまくいくのと、うまく行かないものを説明しており、非常に参考になる。

「我々は専門家でもないし、こういう分野の教育にもなれてはいないが、この分野に関する倫理を教えることは我々の責任である。コンピュータサイエンスの教育者として、この分野の社会的、倫理的問題に関してきちんとした基盤を持たせるようにすることが我々の役目である」と締めくくっている。

2. 歴史を教えるべきであるという意見もある。

これも最近よく主張される考え方である。確かに、歴史的に振り返ると、技術にも繰り返しや復活があるとか、かつて不可能と目されたことが現実のものとなっていたりすることをいくつもの事例から学べるであろう。そして、将来についての洞察、どういう研究が成功し、どういう研究がうまく行かないのかなど一種の先を見る目、技術を見る目が出来ると期待される。ただ、これもなかなか教え方が難しいので、工夫が必要であろう。理科系の学生は歴史というと、そこで逃げてしまう恐れもある。社会科が嫌いで理科系に来ている学生もいる。歴史を歴史として教えることは退屈で、面白くないという印象がある。むしろ、要所要所で、歴史的な事項を少しずつ入れていく程度でよいのではないかと考えている。ある事柄の生まれた背景、その効果などをその事柄とともに伝えることは学生の興味を引き立てる可能性が大きい。つまり、戦記物などのように、個人の事績

やエピソードは面白いが、通史になると途端に面白くないと感じるのと似ている。ただ、個別のエピソードだけでは、歴史の流れがわからないという意見もあるが、少なくともこの科目で大々的に取り上げる性質のものとは思われない。その意見に対しては、年表を配布し、話題になったことを年表から探させ、関連した先行事象やその後の発展などを自分で発見するという指導をすればよいのではないだろうか？ 情報工学・科学だけの年表でなく、一般的な社会経済文化と他の科学技術の歴史を対照してみられる年表を用意するとよい。また、大学によっては、技術史の専門家に依頼して講義を用意している。そういう講義の聴講は推奨するとよいであろう。また、ビデオ素材を活用した教材が作られるとよいと思われる。

3. 作文教育について

この分野に限ったことではないが、米国の企業が就職する学生に求めている資質として、もちろん、専門分野の能力は当然であるが、その他、チームワークを組んで仕事を遂行する、リーダーシップも取れる、コミュニケーションスキル(文章を書く、プレゼンテーションができる、interpersonal communication が円滑にできる)、実際にものを自分の手で作った経験などを求めるという。もともと子供の頃から、コミュニケーションはしっかり教育していると言われる米国で、さらに、この能力を付けて欲しいと大学に要望されているという。文章を書くことについては、J90 策定当時からもしばしば取り上げられ、議論も相当行われた。また、最近は熱心な取り組みの報告もあり、テクニカルプレゼンテーションを科目として設けている事例もあると聞く。これらの動きがだんだんと広まり浸透することが望まれるが、ここでも教育法、教材、教員などの問題があり困難は多い。このリテラシ科目の中だけの問題でもないが、一つの可能性を指摘しておく。

まず「企業が求める能力にコミュニケーションスキルが上げられていることなどの説明、ある程度自分でその能力を磨かなければならない。座学で方法論を伝授されたから、よい文章がすぐ書ける訳ではない。文章を書くことの基本の知識をもとに、実際にトレーニングすることが大事である」というような、方向を指示しておき、あまり身構えることなく日常的に文章を書くのが当たり前であるというような場を作っていく。たとえば、毎回短い報告文を書かせ、なんらかのフィードバックをする。これもこの科目を超えて、講義、実験、演習の種々の場で、常日頃トレーニングをつませることが効果的であろう。これは語学と似ている。教室で習う英語はそれだけで英語が読めたり、話せたりするのではなく、相当量のトレーニングを積まないと身につかないのではないか。講義を聞いてほぼ必要な能力が付く種類の教科と、さらにトレーニングを積むべきものがある。集中トレーニングがよいのか、継続トレーニングがよいのかもものによる。文章表現などはむしろ継続的な(自己)訓練が必要なもので、大学では可能な限り日常的に訓練する場をいつらえることと、その後も自己訓練できる能力を付けてやるのが大事ではないだろうか。この日常の訓練の手段として、電子メールなどネットワーク機能の利用があることを上げておく。

4. リテラシ教育について

リテラシは「読み、書き、そろばん」と言われることが多い。慶応大学の相磯教授は、これに、「はなし」を加えておられる。リテラシに何を含めるかは固定的ではなく、さまざまのとらえ方があっても不思議はない。共通的なことをあげるなら、この基本的な知識と、トレーニングを通して身に付ける基本的技能の2面をもつといえよう。本カリキュラム案では、プログラミングとこの科目をリテラシ科目として位置付けている。プログラミングも多様な要素を含みかつ、技能的な訓練も含む。この科目としてはプログラミングで扱わない部分を考慮した。

5. おわりに

以上、リテラシ科目としての「コンピュータサイエンス序説」なる科目の構成要素と、それらの考え方を述べてきた。もちろん、J97-WGでの熱心な議論を土台にしているが、実際に文章化する段階では、やや担当者の個人的なニュアンスが出ているかもしれないと恐れるが、それを緩和する

ために、この科目の構成はかくあらねばならないという固定的な一つの案を示すことはせず、構成要素として考えられるもの4つの柱をまず提示した。伝統的な概論や序説の考えと違う要素を加えていることに気づかれたであろう。つまり、伝統的なCの概論、あるいはAのリテラシだけでなく、B、Dのような要素も含めた。これは、米国や日本の大学での、従来の基礎から積み上げていくカリキュラムの見直しの流れを考慮したものである。現在の50代以上の教員が大学へ入る頃の大学進学率は10%程度であった。そして、昨今は大学進学率は50%近くになっている。当然、学生の意識も変わっており、また、社会情勢も大きく変化しており、求められる学生像も変化している。単に、この科目は後で必要になるからというガイダンスだけでは学生はまともに学習しない。学習に対する意欲、準備、意味、方法を持たない学生が圧倒的に増えている現実がある。まずは、学習する意欲を喚起し、学習する態度の準備をさせ、方法を伝えることも無視できないのである。言い換えると、動機付けが大事であるという考え方である。動機付けの工夫・試行はいろいろ行われている。米国でうまくいったと称する事例を聞いても、日本でそれをそのまま実行することは、教育資源(人、もの、金、スペース)の制限からとても無理というものも多い。カリキュラム設計者が考えるべきことは、実現可能な、かつ、一過的な実験授業的なことでなく、長続きするものを学科の責任で選び取ることが重要である。その意味で、構成要素と考え方を主に述べた。これをどうアレンジするかは各大学・学科の事情、目標で異なって当然である。また、この科目だけでは扱い切れない重要なことも多い。それを全体のカリキュラムの中でどのように配置していくかも問題で、これはこの科目だけの問題ではない。

L-2 プログラミング入門 [4]

目的

プログラム言語を用いて、手続き型のプログラミングについて学ばせる。プログラム作成の経験を仮定しない。特定のプログラム言語の詳細な文法を教えることが目的ではなく、手続き型言語によるプログラミングの諸概念を学ぶことが目的である。また、以後受講するプログラムあるいはソフトウェア関連科目のガイド的な役が果たせれば好ましい。

概要

問題を解くときに、手続き型プログラムでどのように行うかを学ぶ。手続き型プログラムの諸概念を段階的に学ぶ。間違いの少ないプログラムをどのように開発するかを、プログラミングの歴史とともに学ぶ。

[修得項目]

以下に示す手続き型プログラム言語の諸概念

[知識項目]

整数型と実数型の機械表現・データの精度，コンパイラ言語の構文図，構造的プログラム，goto 文なしのプログラム，デバッグ，プログラムの階層化，段階的詳細化，スタック，抽象データ型，種々の文字コード，時間計算量，空間計算量，

[関連項目]

構文図とコンパイラの再帰呼び出し，ランダムアクセスファイル，ごみ集め，

講義項目

1. コンパイラ言語プログラムの実行機構 [2]

ソースプログラム，出力文，コンパイラ，オブジェクトプログラム

2. 簡単なプログラム [4]

入力文，変数，式，代入，変数宣言，整数型，実数型（整数型と実数型の機械表現・データ）

3. 簡単なループを用いたプログラム [4]

for 文，定数，定数定義，標準手続き・標準関数，型変換，出力指定（コンパイラ言語の構文図）

4. 制御構造 [6]

if 文，while 文，論理式，論理型，アルゴリズムの図式表現（goto 文なしのプログラム，条件式の評価順，デバッグ，NS チャート，PAD，流れ図）

5. 配列 [6]

配列，型定義，配列添字，番兵，多次元配列（配列のメモリ上での実現方法，計算量の概念）

6. 副プログラム [8]

手続き宣言，関数宣言，仮引数，実引数，引数の渡し方，手続き呼び出し，関数呼び出し，ブロック構造，スコープ規則，局所名，大域変数（プログラムの階層化，段階的詳細化，スタック，手続きの実行時状況）

7. 文字列とテキストファイル [6]

文字型，テキスト，行末，ファイル末，文字列型，入出力変換，標準入出力（種々の文字コード）

8. 再帰を用いたプログラム [8]

再帰呼び出し，相互再帰，再帰除去，バックトラック法（分割統治法，構文図とコンパイラの再帰呼び出しの関係，再帰プログラムの時間計算量を求めるための漸化式）

9. ファイル処理 [6]

レコード型，ファイル型，バッファ変数，逐次ファイル，ファイル処理（ランダムアクセスファイル）

10. ポインタを用いたデータ構造 [8]

ポインタ型，動的変数，再帰的データ構造（リスト，木），再帰的处理，ポインタの添字表現（ヒープ領域，ごみ集め，ポインタを用いた抽象データ型の実現方法）

（注）(...) 内は「簡単に触れる」項目．

先修科目

コンピュータリテラシ（ログイン・ログアウト，エディタの使用（OSレベルの）ファイル作成・複製・除去）

後修科目

アセンブラ言語によるプログラミング，プログラミング言語入門

関連科目

データ構造とアルゴリズム，コンパイラ，プログラム設計/構成論，ソフトウェア工学

演習

- 演習は必須である．週に 4 講時とり，2 講時を講義に 2 講時を講義内容の演習にし，通年科目とすることが好ましい．
- 演習時間には，各人に 1 台のワークステーションあるいはパソコンを必要とする．講義時間以外にもこれらの計算機設備を利用できることが好ましい．
- 1 月に 1 つぐらいのプログラム作成をレポート課題とすることが好ましい．
 - － while ループと if 文を用いる課題
 - － 配列を用いる課題
 - － 手続き・関数を用いた課題

- ファイルを用いた課題
 - 再帰を用いた課題
 - 動的変数を用いた課題
- 最初は、講義で説明したプログラムを少し変更するような演習がよいであろう。演習時間には、できあがったプログラムを説明するのではなく、プログラムを開発する試行錯誤の過程を示すのがよい。
 - 内容が進むにつれ、プログラムの変更だけでなく、プログラム全体を最初から作る課題を与えるのもよい。その場合は、プログラム作成に手間取るので、演習時間の比率を高くしてもよい。講義中の適当な時期にヒントを与えたり、宿題提出後にプログラムの開発過程を説明することが必要である。
 - プログラム化すべき課題の仕様を漠然と与え、具体的細部を学生各自に考えさせる要素を残すようにすることも教育効果があがる。その場合は、数人の学生の細部の例を示し、学生と共に検討するとよい。
 - 演習課題で作成するプログラムに入力するデータとして、データ数が非常に多い場合の実行時間を、タイマ機能を用いて計測させることも重要である。また、同一課題を別のアルゴリズムでプログラム化した場合の実行時間の比較も教育効果がある。
 - もし、演習時間をこのように確保できない場合は、週に2講時の講義のみを行い、演習は宿題とすることも可能である。ただし、相当時間の自宅学習が必要となることに配慮すること。

実施方法

- 受講生は、プログラミングの経験がないと想定している。したがって、プログラム言語としては適度に高級な言語（たとえば、PASCAL あるいは C）がよい。アセンブラ言語との対応は、アセンブラ言語の講義で教える。
- 演習のための時間をとれない場合は、講義だけを2半期で行い、演習部分は宿題とすることもできる。
- 各講義項目のありがたみ（その機能がなければどのように不便か）を説明すること。
- デバッグ、構造化プログラミング、抽象データ型の考え方を説明するとともに、関連する高級言語の進化の歴史を教えることも効果的である。
- 履修順で難しいが、「データ構造とアルゴリズム」で学習するアルゴリズムをプログラム化することは効果的である。
- 本科目では手続き型のプログラミングを学ぶが、異なる考え方のプログラミング（たとえば、論理型プログラミングあるいは関数型プログラミング）を2年次あるいは3年次に教えることが望ましい。また、手続き型のさらに進んだ考え方として、オブジェクト指向プログラミングを教えることを勧める。

教科書/参考書

- 言語 PASCAL や C に関する教科書は多く、適当なものを選択すればよい。プログラミングそのものの教科書は少ないので、担当教官が講義中で伝授する。

学部科目 (情報数学系) M

情報代数と符号理論 M-1

離散数学 M-2

計算論 M-3

確率論と情報理論 M-4

数理計画法 M-5

数理論理学 M-6

M-1 情報代数と符号理論 [1]

目的

符号理論の基礎として，群，環，体の代数構造と，その符号理論への応用を教える．具体的には，代数構造として，多項式環，ガロア体などを取り上げる．また，符号理論では，線形符号の構成法や使い方を教える．なお，符号理論に焦点をあててはいるが，基礎理論の学習を通じて代数の美しさや代数的な考え方を経験させることも重要である．

概要

符号理論の数学的基礎である群，環，体の代数構造を学ぶ．さらにその応用として，線形符号の構成法やその使い方を学ぶ．

[修得項目]

代数構造（とくに線形代数）における，諸概念の意味と役割，ガロア体の構成法と算法，基本的な線形符号の設計法，代数的な考え方

[知識項目]

群，環，体の代数構造の違い，誤りと符号間の距離，整数環，イデアル，多項式環，多項式環のイデアル

[関連項目]

アナログ的な信号処理との関係，BCH 符号，リードソロモン符号，情報理論

講義項目

1. 群，環，体の基礎理論 [2]

群，環，体の定義と性質，整数環，イデアル，多項式環，多項式環のイデアル

2. ガロア体 [2]

ガロア体の構成法と表現，共役元と最小多項式，ガロア体の算法，ベクトル空間

3. 符号理論への導入 [2]

符号化と復号，誤りと距離，最尤復号，誤り制御

4. 線形符号の基礎 [2]

生成行列と検査行列，線形符号の復号，部分符号と修正，ハミング符号

5. 巡回符号 [2]

巡回符号の表現，巡回符号の符号化，誤り検出，巡回ハミング符号と誤りトラップ復号法

6. その他の符号 [1]

BCH 符号，リードソロモン符号，情報理論

先修科目

線形代数

後修科目

デジタル通信，確率論と情報理論，論理回路

関連科目

離散数学，画像情報処理，パターン認識

教科書/参考書

- 符号理論，今井 秀樹 著，コロナ社，1990 (教科書)
この科目の講義内容に即した教科書．非常にバランスよく書けている．
- 符号理論入門，岩垂 好裕 著，昭晃堂，1992 (参考書)
上記の教科書とはまた違った見方や話題を提供してくれる本．
- Error-Correcting Codes, W. Peterson and E. Weldon, MIT Press, 1972 (参考書)
符号理論のバイブル的な本．学部生でも積極的な学生には，英語の本を読んでもらうことを勧めてみるべきだろう．

M-2 離散数学 [1]

目的

CSに必要となる数学的な概念，記法，論法の基礎を修得させる．単なる知識の修得を目指すのではなく，考えていることを形式的に記述し議論する手法，また，形式的に記述されていることから，直観的なイメージを得る方法を身につけさせる．ただし，数学的な技法の修得だけに片寄り過ぎると，学生が興味を失ったり，数学嫌いになる恐れがあるので注意が必要（詳しくは実施方法参照）．

概要

CSに必要となる数学的な概念，記法，論法の基礎を修得する．

[修得項目]

集合，関数，論理などに関する基本的な記号の使い方，基本的な数学論法（背理法，帰納法）の使い方，再帰的な定義や証明の方法

[知識項目]

写像の種類，関係，順序，論理演算，論理式，木，グラフ，漸化式，二項係数

[関連項目]

形式的証明法（自然演繹法），正規表現，正則言語，形式文法，ブール代数，木の探索，グラフの連結性，グラフの彩色，数え上げ手法

講義項目

1. 集合，関数，論理などの記号の定義と使い方 [2]

集合の記法，関数に関する諸概念（定義域，値域，全射，単射），命題論理記号，述語論理記号

2. 関係，順序などの基本的な数学概念とその使い方 [2]

二項関係，同値関係，半順序関係，全順序関係

3. 帰納法を使った証明，再帰的な定義などの方法 [3]

帰納法の原理，帰納法の各種バリエーション，再帰的な定義の例，構造に基づく帰納法（再帰的に定義されたものに対する帰納法），各種数え上げ問題

4. グラフ理論の基礎 [3]

木，全域木，木の探索，グラフの連結性，グラフの彩色

先修科目

線形代数，解析学

後修科目

計算論，数理論理学，プログラミング言語論，形式言語とオートマトン，データ構造とアルゴリズム，数理計画法

関連科目

論理回路，情報代数と符号理論，確率論と情報理論

実験及び演習

- 週に一度程度の割合の演習を行なうか，毎回の授業で小テストかレポート提出をさせることが望ましい。

実施方法

- 講義項目は，この科目で教えるべき各項目を並べあげたもので，講義の内容や順序とは無関係である．たとえば，再帰的な定義や帰納法などの修得は，それ自体で行なうのは無意味であり，たとえば木を題材にして教えていくべきだろう．
- どの科目でもそうだが，とくにこの科目は単なる数学技法の修得だけに片寄ると無味乾燥な授業になりやすい．学生の興味を引きつつ教えるには，情報系の様々な話題と関連させたり，パズルなどを題材に用いて，直観的にわかりやすいものから導入するとよいだろう．

教科書/参考書

- コンピュータサイエンスのための離散数学，守屋 悦朗 著，サイエンス社，1992（教科書）
CSの基礎を広く紹介している．話題が多いので，適当に選択して教える必要がある．
- CS入門，M. Arbib, A. Kfoury, R. Moll 著，甘利 俊一，金谷 健一，嶋田 晋 訳，サイエンス社，1984（教科書）
CSの基礎を浅く広く紹介している（ただしグラフ理論に関する話はない．）定義やその簡単な使い方が中心なので，担当教員が肉付けをしていく必要がある．
- グラフ理論入門，齊藤 伸自，西関 隆夫 訳，近代科学社 1994（参考書）
基礎的なことがしっかりと書かれており，訳もしっかりしていて読みやすい．ただし，情報系への応用についてはほとんど言及がない．
- コンピュータの数学，R. Graham, D. Knuth, O. Patashnik 著，有澤 誠，安村 通晃，萩野 達也，石畑 清 訳，共立出版，1993（教員用参考書）
離散数学に関する豊富な話題，例題を提供している（ただしグラフ理論に関する話はない．）

M-3 計算論 [1]

目的

計算の原理，計算モデルを様々な角度から述べ「計算」の本質に対する理解を深めさせる．3種類の計算モデル（計算機械，帰納的関数，ラムダ計算）の同値性を学ぶことにより「計算」には様々な側面があることを学ばせる．また「計算」には限界があること，その限界がなぜ避けられないのか，そしてどのように影響するのか，などを理解させる．

概要

計算の原理，計算モデルを様々な角度から述べ「計算」の本質に対する理解を深める．また「計算」の汎用性や「計算」には限界があることなどを学ぶ．

[修得項目]

計算の異なる表現の仕方（計算機械，帰納的関数，ラムダ計算），データのコード化の意味，計算の汎用性，計算の限界

[知識項目]

インタプリタの概念，計算モデルの同値性，計算不可能性の証明，原始帰納的関数と帰納的関数の違い

[関連項目]

プログラミング・スタイル（手続き型，論理型，関数型），プログラムの意味論，計算の複雑さの解析，その他の計算モデル（例：並列・並行計算モデル）

講義項目

1. 計算機械による計算可能性の議論 [4]

手続き型プログラムによる計算の表現，各種データの0,1-列上へのコード化，インタプリタの構成，計算不可能性の証明

2. 帰納的関数による計算可能性の議論 [3]

帰納的関数，原始帰納的関数，ゲーデル・コード化，インタプリタの構成，計算不可能性の証明

3. ラムダ計算による計算可能性の議論 [3]

ラムダ式，ラムダ変換，ラムダ変換による計算の表現

4. その他の計算モデル [1]

並列・並行計算モデルなど

先修科目

離散数学，プログラミング入門

後修科目

プログラミング言語論

関連科目

形式言語とオートマトン，コンパイラ，数理論理学，データ構造とアルゴリズム

実験及び演習

- 2週間に一度程度の割合の演習が望ましい．演習の代わりに，課題を何度か宿題として与えてもよい．

実施方法

- 最初の「計算機械による計算可能性」を教える際には，先修科目などの状況に応じて2通りの教え方がある．学生にプログラミングの基礎的な知識がない場合には，チューリング機械を用いて教えた方が望ましい．一方，プログラミングの基礎的な知識がある場合には，手続き型プログラムを計算のモデルとして教えた方がよいだろう．
- 講義では，数学的な議論を通して「計算」の本質に対する理解を深めさせることが中心となるが，討論などを通して直感的な理解を深めさせることも必要かもしれない．たとえば，学生同士のグループで自由に議論しながら，計算の基本要素を抽出させるような演習もおもしろいだろう．
- 一見して項目が多そうだが「ラムダ計算による計算可能性の議論」までは十分に教えられるだろう．もし時間的に無理な場合には「計算機械による計算可能性の議論」を多少省略するとよい．この部分は，通常のプログラミングなどを通して体験できることが多いので，多少，急ぎ足で教えることも可能だと思われる．

教科書/参考書

- 計算論入門，米崎 直樹，渡辺 治，日本評論社，1997（教科書）
この科目の講義内容に即した教科書．プログラム言語モデル，チューリング機械のどちらからでも入れる構成になっている．並列・並行計算のモデルについても簡単な解説がある．
- 計算論 — 計算可能性とラムダ計算 —，高橋 正子 著，近代科学社，1991（参考書）
前半は，この科目の講義内容に完全に即した内容．一方，後半は，ラムダ計算とラムダ・モデル理論へのすぐれた入門書になっている．しかし，全体にレベルが高く，学部レベルの教科書としては重すぎる．
- Mathematical Theory of Computation, Z. Manna, McGraw-Hill, 1974（参考書）
この科目の講義内容からは多少はずれるが，名著であり，また英語でも読みやすいので，意欲のある学生に勧めるとよいだろう．
- 計算可能性入門，小林 孝次郎 著，近代科学社，1980（教員用参考書）
この科目の講義内容に，ほぼ即した内容．残念ながら絶版になっている．

備考 (コメント)

- プログラムは計算の指令書であり，手続き型プログラムを計算機械モデルと考えるのは正確にはおかしい．しかし現代では，常識的なプログラミング言語に対しては，その裏にある機械まで含めて常識的な意味解釈が成り立っており，そこまで含めればプログラムを計算機械モデルと見なすことができる．

M-4 確率論と情報理論 [1]

目的

確率の基礎概念と、情報分野においてよく利用される確率の手法の原理を理解させ、確率を利用した計算法を修得させる。内容的には、確率論などの名称で従来教えられていた項目のうち、離散的な事象に関する確率に絞って教える。後半では、確率の応用分野として、情報理論へ話しを進め、情報理論の考え方や基本原理を教える。

概要

確率の基礎概念と、情報分野においてよく利用される確率の手法の原理を理解し、確率に関する計算法を修得する。後半では、情報理論へと進み基本原理を学ぶ。

[修得項目]

確率論の諸概念(条件付き確率, 独立性, 期待値, 分散), 確率の各種計算手法(確率, 期待値, 分散などの評価法), 情報量の意味, 通信モデル

[知識項目]

確率の定義, 確率変数, 期待値, 分散, 確率分布(二項分布, ポアソン分布, 正規分布), 確率の法則(マルコフの不等式, チェビシェフの不等式, 大数の法則), クラフトの不等式, ハフマン符号, シャノンの第一定理相互情報量, 通信路容量, シャノンの第二定理

[関連項目]

ランダムさの定義, コルモゴロフ複雑度, 収束定理(中心極限定理)など, 符号理論

講義項目

1. 確率の基本概念 [2]

確率の定義, 諸数値(確率, 期待値, 分散)の定義と求め方

2. 確率分布 [2]

二項分布, ポアソン分布, 幾何分布, 正規分布

3. 確率の法則 [1]

確率の法則(マルコフの不等式, チェビシェフの不等式, 大数の法則など),

4. 情報の符号化と情報量 [3]

問題の形式化, 情報量, クラフトの不等式, ハフマン符号, 算術符号, シャノンの第一定理

5. 通信と情報量 [2]

通信のモデル化, 相互情報量問題, 通信容量, シャノンの第二定理

先修科目

解析学, 離散数学, 情報代数と符号理論

後修科目

デジタル通信，信号処理，画像情報処理，パターン認識

実施方法

- 確率論，情報理論，ともにややもすると抽象的な話しになりがちなので，具体例を多く取り入れるようにした方がよい．たとえば，確率の法則のところでは，故障して誤った答えを出す可能性のあるチップを複数個使い，誤り確率を減らす方法の効果を解析するなど，情報系の学生の興味に即した話題を提供すると効果的だろう．同様に，情報理論のところでも，符号やデータ圧縮との関連を中心に説明するとよい．
- 確率分布や確率の法則は，情報理論との結び付きが弱いので，とくに情報理論に進む前に教える必要はない．確率の基礎概念の後，すぐに情報理論に進んでもよいだろう．ただし，確率分布や確率の法則の基本的な知識は，様々な分野で必要となってくるので，安易に省かない方がよい．
- 教育課程の性質や目標によっては，後半の題材に統計を持ってくるといいう，従来型の選択もある．ただし，その場合でも，代数系と符号理論など，別の授業科目で情報理論を教えるべきである．

教科書/参考書

- 確率，情報，エントロピー，有本 卓，森北出版，1980（教科書）
この科目の講義内容にかなり近い．教科書としては，やや高度かもしれない．
- 自然科学の統計学，東大教養部統計学教室編，東大出版会，1987（参考書）
この科目の設計では，統計までふれられないので，統計についてよい解説書を勧めておくとよいだろう．
- 確率論とその応用 I（上），W. Feller 著，河田 龍夫 監訳，紀伊国屋書店，1993（参考書）
確率論の定版ともいえる教科書．ただし，学部レベルの教科書として使うには適さない．
- Randomized Algorithms, R. Motwani and P. Raghavan, 1995（教員用参考書）
最近，アルゴリズムの開発や解析で，確率的な議論が多く利用されるようになっているが，本書はそのような話題を中心に集めた教科書．学生に具体的な話題を提供するための参考書に使うとよいだろう．

M-5 数理計画法 [1]

目的

世の中の多くの問題は最適化問題として定式化できることが多い．本科目では，最適化問題を解く手法として数理計画法の基礎を理解させ，それを応用できる力を身につけることを目的とする．とくに，問題をうまく定式化する手法を修得させ，定式化された問題を解く際に，適切なアルゴリズムを選択し（設計し）使うことの重要性を認識させることが重要である．

概要

最適化手法として数理計画法の基礎を理解し，それを応用できる力を身につけることを目的とする．

[修得項目]

数理計画問題への定式化の手法，代表的な解法の概要，問題分析における優れたアルゴリズムの有用性

[知識項目]

双対定理，シンプレックス法，ネットワーク最適化法最適条件，勾配法，

[関連項目]

内点法，動的計画法，分枝限定法，アルゴリズムの設計，計算の複雑さ

講義項目

1. 線形計画問題への定式化 [1]

線形計画問題，線形計画問題への定式化，可能解領域

2. 線形計画問題の解法 [2]

シンプレックス法，シンプレックス法のバリエーション，双対定理，内点法

3. 特殊な線形計画問題とその解法 [2]

輸送問題，ネットワーク・フロー問題，整数計画問題

4. 非線形計画問題 [3]

未定乗数法，最適条件，勾配法

5. その他の最適化手法 [2]

動的計画法，分枝限定法

先修科目

線形代数，解析学，離散数学，プログラミング入門

後修科目

人工知能

関連科目

データ構造とアルゴリズム，数値計算

実験及び演習

- 適当な規模の線形計画問題を解くプログラムの作成は，プログラミング実習/コンピュータ実験のよい題材になるだろう．また，多少大きな問題の分析を，様々な（フリー）ソフトウェアパッケージを使いながら行うという課題も，ソフトウェア利用環境の実習課題として有意義だろう．

教科書/参考書

- 数理計画，刀根 薫，朝倉書店，1994（教科書）
数理計画に関する話題が，コンパクトにまとめられている．わかりやすい．
- 数理計画法，一森 哲男 著，共立出版，1994（教科書）
この科目の講義内容にほぼ即した教科書．多くの話題がコンパクトにまとめられている．
- 線形計画法，今野 浩，日科技連，1987（教員用参考書）
事例が多いので，例題や動機付けに利用できる．
- 数理計画モデルの作成法，H. P. Williams，小林 英三 訳（前田 英次郎 監訳），産業図書，1995（教員用参考書）
数理計画法を実際に応用することを目標とした本であり，モデルの作り方などから書かれている．また，XPPRESS-MP パッケージ用のプログラムも用意されているので，この本を参考に実験を計画することもできるだろう．

目的

計算機の論理回路の理論的基礎であり，論理的思考の基礎でもある命題論理，および第1階述語論理の構造および証明法の基本事項の習得を目的とする．論理学の基本事項として，構文論，公理系，意味論，モデル，公理系の完全性，代表的な証明法，証明法の完全性・健全性などの概念を正しく習得させる．講義の進め方としては大きく2つの方向が考えられる．一つ目は論理式の定義や公理系から出発し，意味論，公理系の完全性および構文論と意味論の一致，代表的証明法（自然演繹法あるいは導出原理）など理論的な基礎について講述し，余裕があれば高階論理や様相論理に言及する方向である．もう一つは，公理系やその完全性についての議論は簡単にすまし，意味論を中心に講述し，導出原理およびそれに関わる様々な戦略の理解を目標に講義を進める方向である．後者の方向では必要に応じて論理プログラムの基礎に言及することも好ましい．

概要

命題論理，第1階述語論理をはじめとする古典論理の基礎について講述する．構文論と意味論の関係，完全性や健全性などの基本概念を理解させる．また，証明系として，自然演繹法あるいは導出原理など代表的な系と関連する概念を説明する．

[修得項目]

論理演算子，論理式の真偽値，真偽値表，構文論と意味論，限量子，論理式の解釈，モデル，積和標準形，節表現，単一化，導出原理の基本操作

[知識項目]

命題論理・第1階述語論理の完全性と健全性，Modus ponens，エルブラン領域，自然演繹法，導出原理の完全性と健全性，スコーレム標準形，単一化アルゴリズム，エルブラン解釈，持ち上げ補題，導出原理の完全性と健全性，導出原理の種々の戦略，論理プログラム

[関連項目]

高階論理，様相論理，内包論理，非単調論理

講義項目

1. 命題論理 [2]

命題変数，論理演算子，命題論理式，論理式の真偽値，真偽値表，命題論理の公理系，構文論と意味論，命題論理の完全性と健全性

2. 第1階述語論理の構文論と意味論 [4]

第1階述語論理の公理系，Modus ponens，論理演算子，限量子，意味論，論理式の解釈，モデル，完全性と健全性，エルブラン領域

3. (自然演繹法 [1])

推論規則，自然演繹法

4. 導出原理 [6]

積和標準形，節表現，命題論理の導出原理，導出原理の完全性と健全性（命題論理版），第 1 階述語論理式の節表現，スコーレム標準形，代入，単一化，単一化アルゴリズム，エルブラン領域，エルブラン解釈，持ち上げ補題 (lifting lemma)，導出原理の完全性と健全性（第 1 階述語論理版）

5. 導出原理の戦略 [2]

意味導出法，入力導出法，線形導出法，SL 導出法，種々の戦略の完全性の証明

6. その他の論理体系 [1]

高階述語論理（第 2 階述語論理），様相論理，内包論理，非単調論理

7. 論理プログラム [1]

Horn 節，Prolog の構文と実行方式，プログラム例，不動点意味論

先修科目

離散数学

後修科目

人工知能

関連科目

形式言語とオートマトン，論理回路，自然言語処理

実験及び演習

- 単一化アルゴリズムのプログラム作成
- 任意の第 1 階述語論理式を Horn 節集合に変換するプログラム作成
- 導出原理による定理証明プログラムの作成

実施方法

- 任意のプログラミング言語で上記のアルゴリズムを実現する，あるいは，Prolog 言語の実習を兼ねて上記プログラムを Prolog で実現する．

教科書/参考書

- 情報の論理数学入門 - ブール代数から述語論理まで，小倉久和，高濱徹行，近代科学社，1991（教科書）
情報の学生が論理学を学ぶための教科書．
- 述語論理と論理プログラミング，有川節夫，原口誠，オーム社 知識工学講座 4，1988（参考書）
述語論理の基礎と論理プログラミングの理論的基礎について簡潔にまとめている．
- 人工知能基礎論，M. R. Genesereth, N. J. Nilsson 共著 古川康一 監訳，オーム社，1993（参考書）
人工知能の分野における論理的基礎が過不足なくまとめられている．
- 情報科学における論理，小野寛晰，日本評論社，1994（教員用参考書）

学部科目 **U**

- 論理回路 **U-1**
- 形式言語とオートマトン **U-2**
- データ構造とアルゴリズム **U-3**
- コンピュータアーキテクチャ **U-4**
- プログラミング言語論 **U-5**
- 論理設計 **U-6**
- オペレーティングシステム **U-7**
- コンパイラ **U-8**
- デジタル通信 **U-9**
- データベース **U-10**
- 人工知能 **U-11**
- 情報ネットワーク **U-12**
- ソフトウェア工学 **U-13**
- 数値計算 **U-14**
- 集積回路工学 **U-15**
- 信号処理 **U-16**
- 画像情報処理 **U-17**
- パターン認識 **U-18**
- ヒューマン・コンピュータ・インタラクション **U-19**
- コンピュータグラフィックス **U-20**
- 自然言語処理 **U-21**

U-1 論理回路 [1]

目的

計算機科学の基礎の一つであるブール代数とその電気回路的な実現である論理回路の関係を、数学的概念と物理実現の対応として教える。具体的には、情報と電気信号の対応、論理関数の組み合わせ論理回路による実現、有限状態機械の順序回路による実現を理解させる。効率の良い設計手法は論理設計に譲り、ここでは論理回路によって数学的概念（論理関数や有限状態機械）が実現できる原理を教えることに専念する。情報科学の諸概念と物理的な実現のつながりをできるだけ体験的に実感させることを狙う。

概要

デジタル回路の基本となる論理回路の諸概念を理解する。アナログ信号とデジタル信号の違いや情報のデジタル符号化の意味を理解させる。また、組合せ回路によって論理関数を実現されること、および順序回路により有限状態機械が実現されることを理解させる。

[修得項目]

アナログとデジタルのちがい、物理量と情報の対応、離散化の意味、論理関数の定義、論理関数の基本性質、論理関数と組み合わせ論理回路の対応、基本論理素子の構造と原理、記憶素子の構造と原理、有限状態機械の定義、有限状態機械と同期式順序回路の対応

[知識項目]

論理式の性質と変換規則、論理関数の表現法、真理値表、カルノー図、状態遷移表、状態遷移図、同期式順序回路と非同期式順序回路、基本演算回路、Moore 型と Mealy 型、数表現、加算器の実現

[関連項目]

論理関数の種類、論理関数のキューブ表現、ハードウェア記述言語、各種フリップフロップ、信号の離散化と時間の離散化、NPN 同値、双対性、2分決定図、カウンタ、セレクタ、シフタ

講義項目

1. アナログとデジタル [1]

アナログとデジタルのちがい、物理量と情報の対応、信号の離散化、時間の離散化

2. ブール代数の基礎 [3]

ブール代数の基礎、代数の基本概念（同値関係など）、基本論理演算、論理関数と論理式、真理値表、NPN 同値、双対性

3. 論理関数とその表現 [3]

論理式、積和表現、和積表現、キューブ表現、幾何学的イメージ、2分決定図、ハードウェア記述言語、表現の評価の尺度、関数族

4. 組み合わせ論理回路 [2]

基本論理素子（各種ゲート）の構造と原理（CMOS）、組み合わせ回路による論理関数の実現

5. 有限状態機械 [2]

有限状態機械の定義，有限状態機械の例（自動販売機など），内部状態，状態空間，次状態関数，出力関数，Moore 型と Mealy 型

6. 同期式順序回路による有限状態機械の実現 [2]

記憶素子の構造（フリップフロップ），次状態関数と出力関数の実現

7. 計算機の基本回路 [2]

数表現，加算器の実現（直列加算器，順次桁上げ加算器，桁上げ先見加算器），カウンタ，セレクタ，シフタ

先修科目

形式言語とオートマトン，ハードウェアの構成（コンピュータサイエンス序説），離散数学，情報代数と符号理論，

後修科目

論理設計，集積回路工学

関連科目

コンピュータアーキテクチャ，データ構造とアルゴリズム

実験及び演習

- 論理回路の製作実験：簡単な論理回路（カウンタ，加算器，シフタなど）を試作させる．小規模な集積回路（SSI）を使って，できるだけ物理的な回路に触れさせる機会を持たせる．電源やアースの配線も体験させて，電気回路として実現されていることを理解させる．

教科書/参考書

- 論理回路理論，山田輝彦，基礎情報工学シリーズ，森北出版，1990（教科書）
論理回路の性質から合成法までわかりやすく説明されている．
- 論理回路，高木直史，New Text 電子情報系シリーズ，昭晃堂，1997（教科書）
古典的な論理回路に関する事項をコンパクトにまとめて平易に解説している．
- 論理設計スイッチング回路理論，笹尾勤，近代科学社，1995.（教科書）
論理関数の性質について詳しい．理論的に良く整理された教科書である．
- Switching and Finite Automata Theory, Second Edition, Zvi Kohavi, McGraw Hill, 1978.（教科書）
明快な英語と整理された記述の論理関数や論理回路に関する古典的教科書．英語の勉強にもよい．
- 論理回路の基礎，田丸啓吉，工学図書，1989.（教科書）
論理回路設計の基礎を平易に解説している．
- 論理回路工学，久津輪敏郎 江端克彦，共立出版，1994.（教科書）.

U-2 形式言語とオートマトン [1](または [2])

目的

情報工学/科学の基礎となる内容である。直接この議論を必要とするコンパイラに関する科目も有り得るし、CSの多くの分野で、形式言語とオートマトンの考え方や用語、結果は当然の知識として使用されている。したがって、多くの分野で、研究者技術者としての基本的な素養という位置を占めている。また、厳密な論証と直感的なモデルの構築など、抽象的思考を主と知的訓練をねらう。

概要

この科目の内容は syntax を扱う基本的な方法であり、CSの多くの分野に必須の共通基礎知識といえるものである。理論的な科目ではあるが、興味深い内容を多く含む。練習問題をできるだけ解くことでその楽しみを味わうことができるであろう。

[修得項目]

有限オートマトン，状態遷移図，非決定性オートマトン，正規表現，クリーネの定理，正規文法/言語，文脈自由文法/言語，文脈依存文法/言語，句構造文法/言語，Chomsky の階層，導出木，プッシュダウンオートマトン，文脈自由文法/言語の諸性質，チューリングマシン，万能チューリングマシン，停止問題，決定不能問題，複雑さ（計算量）の概念

[知識項目]

正規言語と文脈自由言語の Pumping Lemma，Chomsky の標準形 (CNF)，Post の対応問題，P と NP

[関連項目]

UNIX ツールで用いられる正規表現，PGA での順序回路の実現，Myhill-Nerode の定理，構文解析とコンパイラ，プログラム言語の文法，自然語と人工言語，自然語のシンタックスは扱えるか，線形拘束オートマトン，Hilbert の第 10 問題，Church の提唱

講義項目

1. 準備（もちろん，他の科目で既習のものは省略できる。）

集合，n 項組，関数，アルファベット，ストリング，述語，量化記号，証明法（背理法，数学的帰納法）

2. 正規言語

有限オートマトン，非決定性オートマトン，閉性，正規表現，クリーネの定理，*Pumping Lemma と応用，**Myhill-Nerode の定理と応用

3. 文脈自由言語

文脈自由言語，導出，導出木，正規文法と有限オートマトン，CNF，Pumping Lemma とその応用，閉性（一部 *，**の内容あり），プッシュダウンオートマトン，文脈自由言語とプッシュダウンオートマトンの関係，*決定性プッシュダウンオートマトン，**構文解析とコンパイラ，**プログラム言語の文法

4. 文脈依存言語

チョムスキの階層，*線形拘束オートマトン，**閉性，**文脈依存言語の諸性質，*線形拘束オートマトンとの関係

5. チューリングマシン

万能チューリングマシン，停止問題，*非決定性チューリングマシン，*チューリングマシンの各種変形

6. 決定問題

Post の対応問題，**Hilbert の第 10 問題，正規言語，文脈自由言語の決定問題，**文脈依存言語の決定問題

7. *複雑さ

P と NP，NP 完全問題

先修科目

離散数学，プログラミング入門

後修科目

コンパイラ，人工知能，プログラム意味論

関連科目

計算論，数理論理学，論理回路，データ構造とアルゴリズム，プログラミング言語論

実験及び演習

- この科目では，できるだけ多数の演習問題を解くことが効果的である．時間が許せば対応する演習を行うとよい．

実施方法

- 上の項目は標準的な教科書の項目である．これを演習つきで丁寧にやろうとすると，2 単位 (1 半期) では苦しい．従って，詳しく丁寧にやる必要性を感じる場合は 4 単位 (2 半期) とし，やや基礎概念を一通りざっとみておくのでよいということであれば，2 単位科目として構成すればよい．また，すべてを教え込む必要は必ずしもなく，むしろ基礎を丁寧に教え，この種の考え方や議論の進め方を理解してもらっておく方がよいともいえる．そのあと，足りないところは大学院に残りの部分の内容をカバーする科目を設けるなどして，後続の科目や参考書の自習などができる程度になればいい．また，この科目は実は予備知識はほとんどなくてもいいが，実際に学習するにはある程度の学生の成熟 (マチュリティ) が必要であり，数学的な予備訓練，プログラミング言語の経験なども有用である．実際には 3 年生以降が適当であり，そのときに同時に開講される科目が実際には多いと考えて，関連科目としてあげるものを増やした．ただ，これはかなり標準的と思われる内容配列であるが，決定問題に重点をおいて教授するという考え方もあるし，教授者の考えで効果の上がる内容を選択していただければよい．いくら要領よく書かれた教科書を使っても，2 単位で多くのことを扱うのは現在の忙しすぎる学生には過負担になる危険性がある．

教科書/参考書

この科目に対応する教科書/参考書は相当多くある．すべてを調べあげたものではないことをお断りしておく．

(教科書)

- [1] J. E. Hopcroft and J. D. Ullman 著，野崎昭弘，木村泉訳：言語理論とオートマトン，サイエンス社 1971.
定評のある教科書．
- [2] 本多波雄：オートマトン・言語理論，コロナ社 1972.
長年教科書として使われた．
- [3] 福村晃夫，稲垣康善：オートマトン・形式言語理論，岩波書店 1982.
コンパクトにまとめられている．
- [4] 小林孝次郎，高橋正子：オートマトンの理論，共立出版 1983.
丁寧に書かれている．
- [5] 富田悦次，横森貴：オートマトン・言語理論，森北出版 1992.
よく整理されている．
- [6] 都倉信樹：オートマトンと形式言語，昭晃堂 1995.
文脈自由言語までに限定し，演習問題を多数含む．
- [7] 小倉久和：形式言語と有限オートマトン入門，コロナ社 1996.
コンパクトで概観が得られる．
- [8] Rayward-Smith：コンピュータサイエンスのための言語理論入門，共立出版．
手頃な教科書．

(参考書)

多くは大学院生，教員用である．

- [1] M. L. Minsky 金山裕訳：計算機の数学的理論，近代科学社 1970.
無限オートマトンの部分の内容はきわめて魅力的である．
- [2] A. V. Aho and J. D. Ullman: "The Theory of Parsing, Translating, and Compiling I, II, Prentice-Hall 1972,1973.
前半の文法の説明はこの科目の参考になる．
- [3] R. Y. Kain 著，宮川洋，原島博訳：オートマトンと言語理論，産業図書 1978.
- [4] J. E. Hopcroft and J. D. Ullman 著，野崎昭弘他訳：オートマトン，言語理論，計算論 I, II, サイエンス社 1984.
教科書 [1] より内容は新しいが，4年か大学院程度.
- [5] H. R. Lewis and C. H. Papadimitriou: Elements of the Theory of Computation, Prentice-Hall 1981.
- [6] 有川節夫，宮野悟：オートマトンと計算可能性，培風館 1986.

[7] 嵩忠雄，都倉信樹，谷口健一：形式言語理論，コロナ社 1988.

[8] M. D. Davis, R. Sigal and E. J. Weyuker: Computability, Complexity, and Languages, Fundamentals of Theoretical Computer Science 2nd Ed., Academic Press, 1994. ISBN0-12-206382-1. ハンドブック.

備考 (コメント)

この科目は知的訓練に使えると述べたが，もう少し説明しておく．

この科目はアルゴリズムの理論や論理学等との関連も深く，抽象的な思考能力の育成のはじめに使える科目であり，この科目をしっかり学習してもらうとあとのそれらの科目に自然につながる．

抽象的理論的な内容であるが，ある種の直観的モデルも存在し，適度に抽象度を変えて物事を見ることが重要なソフトウェア設計等の能力にもつながるものがある．直観的に理解したことでその証明を形式的にするにはきちんとした論証が必要であるから，綿密な論証をするという習慣や態度が養成できるように指導すると有用である．

これはかつての幾何学が果たした役割を果たすといえる．つまり，理解しやすい問題が設定できるが，その難易度は巾広い．解けたかどうかは比較的明確に判断できるので解決した喜びを味わえ，問題解決の力を付けるのに有用である．しかし，解がひとつとおりに限定されることはほとんどなく，工夫やアイデアが思いもかけない別解をもたらすことがあり，そういう意味でも知的興味をかき立て，かつ，知的な能力を練磨するにも役立つ，

その意味で練習問題を多数やらせることが有用である．易しいものから順にレベルの上がるような練習問題を解かせると力がつくし，身に付くものも大きい．定義・定理・証明の連続という教科書にそのままそっての教授法では学生は飽きてしまうし，身につくものは少ない．

なお，以上の項目列挙では，時間の制約がある場合や演習重視であまり種々の事柄を教え込むという必要のないという場合に省略してもよい内容の候補として，*，**をつけている．無印のものは，修得項目，*印は知識項目，**は関連項目（余力のある時の知識項目）に対応する．これらの*，**のついた項目は全くふれる必要のないものもあるし，「こういう話もあるが，それはあとでのお楽しみ」（他の上級の科目や特論，自学自習）ということで，詳しい話に入らないですますということでもよい．ただ，先生から「こういうおもしろい話があるのだが，この科目では扱わない」といわれると意外とその言葉が頭に残っているという効果もある．この種の科目は一見数学的に見えて，学生にうまく動機づけしなるとなぜこのようなことを学ぶ必要があるのかと思わせる．ここは教員のしっかりした考えが必要なところである．もちろん，導入に最初から数学的な事を持ち出さず，それまでのプログラム言語のシンタクスの記述法から話を始めても決して無駄ではない．BNFでの記述に感激する学生もいる．そういうことから，コンパイラなどの設計に必須の知識であるとか，AIやその他の教科書や文献を示して，それらの用語や知識が当然の前提として用いられていることなどを説明することも効果がある．また，当初動機づけに注意しても本論に入っていくと，技術的な話に限定してしまって，学生があるところでつまずくとそれ以後脱落する事もあり得る．そのような感じのときは一度それまでの議論を簡単に要約し，今どういう議論をしているのかを再確認するという事も学生にとってきわめて有用である．正規表現は（少々変形したものが）UNIXでよく用いられており，現実との関連を述べるのも効果的であろう．なお，最近発行された教科書[渡辺治・著：計算論入門]のまえがきもこのような科目の立場が明確に述べられている．

U-3 データ構造とアルゴリズム [1]

目的

様々なデータ構造と、それを取り扱う基本的なアルゴリズムを通して、アルゴリズムの設計や解析をするための基本的な手法を修得させる。その基本手法を応用できる能力を養うことを目指す。特定のプログラム言語での実現を目指すものではない。

概要

リスト、スタック、木などの基本的なデータ構造の利用方法・実現方法を学ぶ。配列やポインタなど実現方法の違いにより、効率が異なることを学ぶ。種々の優れたアルゴリズムの考え方を通し、問題の本質の捉え方・攻略法を学ぶ。入力データ数をパラメータとする効率評価法概念、その解析法を学ぶ。

[修得項目]

基本的なデータ構造とその利用法、アルゴリズム設計のセンス

[知識項目]

リスト、スタック、探索木、ハッシュ法、アルゴリズムの計算量とその評価法、その他のアルゴリズムの設計手法(グラフ・アルゴリズム、文字列照合アルゴリズム)

[関連項目]

計算の複雑さの解析

講義項目

1. アルゴリズムと計算量 [1]

時間計算量, 領域計算量, 一様コスト基準, 対数コスト基準, O記法, 記法, 記法

2. 基本データ構造 [1]

リスト, スタック, キュー, 木, 抽象データ型

3. アルゴリズム設計技法 [2]

再帰法, 分割統治法, バランス法, 動的計画法, バックトラック法

4. 探索アルゴリズム [2]

2分探索, 2分探索木, 最適2分探索木, 平衡木, ハッシュ, 選択

5. ソーティング・アルゴリズム [3]

選択法, 挿入法, クイックソート, ヒープソート, マージソート, ビンソート, 基底法, 最悪時計算量, 平均時計算量, 計算量の下界

6. グラフ・アルゴリズム [3]

グラフ探索, 2連結成分, 最小生成木, 最短路

7. パターン照合 [2]

正規表現パターン認識, 部分列認識

8. 本質的に計算量の多い問題に対するアルゴリズム [1]

{ NP完全問題, 貪欲アルゴリズム, 近似アルゴリズム, 確率アルゴリズムにも簡単にふれる }

先修科目

プログラミング入門, 離散数学

後修科目

プログラミング言語入門, コンパイラ, データベース, オペレーティングシステム, 人工知能, コンピュータグラフィックス, 自然言語処理, 関数型プログラミング, オブジェクト指向プログラミング

関連科目

数理計画法, 形式言語とオートマトン, 計算論, 論理設計, 論理回路, ヒューマンコンピュータインタラクション

演習

- 演習時間をもうけ, 学んだアルゴリズムを具体的にプログラム化させる.
- 同じ問題を解くアルゴリズムいくつかに関して, 理論的な時間計算量の解析と実際にコンピュータで実行したときの計算時間がどのように関連するかを確認させる演習を行う.
- それぞれのアルゴリズムを理解することは大事であるが, 効率のよいアルゴリズム設計のための共通的な考え方を学ぶことがより重要である. 演習課題もその共通的な設計法を使うことを考慮したものとする.

教科書/参考書

- 石畑 清: アルゴリズムとデータ構造, 岩波書店, 1989. (教科書)
広範囲の内容が非常に丁寧に説明されている.
- 浅野哲夫: データ構造, 近代科学社, 1992. (教科書)
各種のデータ構造とそれを扱うアルゴリズムがコンパクトにまとめられている. 幾何データを扱うためのデータ構造, 整数データを扱うためのデータ構造など, 目的に応じたデータ構造の解説がある. C 言語によるプログラム例が付いているので, 演習にも使えるだろう.
- 茨木俊秀: データ構造とアルゴリズム, 昭晃堂, 1986. (教科書)
学部レベルのすぐれた標準的教科書である.
- R. Sedgewick: Algorithms (2nd ed.), Addison-Wesley, 1988. (参考書)
直観的な図を多く用いていて説明がわかりやすい. アルゴリズム記述は Pascal を用いているが, 最近同じ内容で C を用いたものと C++ を用いたものも出版されている.
- D.E. Knuth: The Art of Computer Programming, Vol.1 ~ Vol.3. (2nd ed.), Addison-Wesley, 1973. (教員用参考書)
さまざまなアルゴリズムを深く検討している.

- A.V. Aho, J.E. Hopcroft, J.D. Ullman: The Design and Analysis of Computer Algorithms, Addison-Wesley, 1974. (教員用参考書)
アルゴリズムのパラダイムに関する代表的な書である。
- M.R. Garey, D.S. Johnson: Computers and Intractability —a Guide to the Theory of NP-completeness—, W.H. Freeman and Company, 1979. (教員用参考書)
NP 完全問題を広く解説し, NP 完全問題のリストが載っている。

コメント

- 将来専門とするコースに関連する題材を選ぶことができると非常に好ましい。例えば「データベース」ではB木など「人工知能」では発見的方法やゲーム木探索など。

U-4 コンピュータアーキテクチャ[2]

目的

「コンピュータアーキテクチャの設計」とは、「コンピュータシステムを構成するハードウェア機能とソフトウェア機能との適切な機能分担(トレードオフ)を図ること」である。従って「コンピュータアーキテクチャ」の講述では、「コンピュータハードウェア」だけに偏ることなく、「ハードウェアとソフトウェアとのインタフェース」に焦点を当てるのが望ましい。すなわち、コンピュータシステムを構成する際のハードウェアとソフトウェアとのインタフェースの設計方式について詳解することが必要である。特に、ハードウェア機構とシステムプログラム(コンパイラやオペレーティングシステムなどの基本ソフトウェア)との関係についても具体的に言及するのが望ましい。「どのようなハードウェアと、どのようなソフトウェアを、どのように組み合わせると、どのようなコンピュータシステムができあがるのか?」についての知識を修得するのが「コンピュータアーキテクチャ」科目の目標である。その点で、本科目で学習する項目はハードウェア及びソフトウェアの各科目で修得できる知識と密接に関連する。

概要

コンピュータシステムにおけるハードウェア機構とソフトウェア機能とのトレードオフ設計をコンピュータアーキテクチャ設計ととらえる立場から、基本的なコンピュータアーキテクチャについて考察を加える。

[修得項目]

コンピュータシステムとアーキテクチャ, ノイマン型コンピュータ, 2進数, プログラム内蔵, 命令とデータ, マシン語, データ形式とメディア, ハードウェア構成, プロセッサ(処理装置, CPU), レジスタ, メモリ, ファイル装置, 入出力装置, 物理的実現(ハードウェア)と論理的実現(ソフトウェア), ハードウェアとソフトウェアのトレードオフ, マシン命令形式, マシン命令の実行サイクル, 基本命令セット, データの属性, 基本データ型, 10進数と2進数, 固定小数点数表現(整数表現), 負数の表現, 浮動小数点数表現, 固定小数点数(整数)の加減算, 固定小数点数(整数)の乗除算, 論理演算, 制御方式, 命令実行サイクルと制御, 実行順序制御, 割り込み, メインメモリ, 補助メモリ, メモリ階層, 仮想メモリ, キャッシュ, 入出力割り込み, コンピュータネットワーク

[知識項目]

コンピュータ技術の歴史(コンピュータの世代), アドレス修飾, データパスとハードウェア構成, 2進化10進数(BCD), 文字コード, 加算の高速化, 浮動小数点数の演算, シフト演算, 10進数演算, 実行の高速化, パイプライン処理, メモリ素子, メモリアーキテクチャ, メディアと入出力, 入出力装置, 入出力方式, 入出力制御, 通信装置, 通信方式, 通信制御, ネットワークアーキテクチャ, LAN, コンピュータ間通信, プログラミング言語/コンパイラ/OS/ハードウェアのトレードオフ

[関連項目]

電子回路, 論理回路(組み合わせ回路, 順序回路), 論理素子, VLSI設計, 有限オートマトン, CAD, ハードウェア記述言語, 論理合成, ハードウェア-ソフトウェア協調設計(コデザイン), システムプログラムの変遷, プログラミング言語, コンパイラ, オペレーティングシステム(OS), ハードウェアアルゴリズム, 計算モデル, コードスケジューリング, プロセス管理, 機能メモリ, 分散処理, ヒューマン-コンピュータ・インタラクション, マルチメディア処理

講義項目

1. コンピュータアーキテクチャとは? [2]

コンピュータシステムとアーキテクチャ, コンピュータ技術の歴史(コンピュータの世代), ノイマン型コンピュータ, コンピュータのハードウェア構成, ハードウェアとソフトウェアのトレードオフ

2. 基本命令セットアーキテクチャ[5]

マシン命令形式, アドレス修飾, マシン命令の実行サイクル, 基本命令セット, データの属性, 基本データ型, データパスとハードウェア構成

3. 数とデータの表現 [3]

10進数と2進数, 固定小数点数表現(整数表現), 負数の表現, 浮動小数点数表現, 2進化10進数(BCD), 文字コード

4. 演算アーキテクチャ[4]

固定小数点数(整数)の加減算, 加算の高速化, 固定小数点数(整数)の乗除算, 浮動小数点数の演算, 論理演算, シフト演算, 10進数演算

5. プロセッサアーキテクチャ[4]

制御方式, 命令実行サイクルと制御, 実行順序制御, 実行の高速化, パイプライン処理, 割り込み

6. メモリアーキテクチャ[4]

メモリ素子, メインメモリ, メモリ階層, 仮想メモリ, キャッシュ

7. 入出力アーキテクチャ[2]

メディアと入出力, 入出力装置, 入出力方式, 入出力制御, 入出力割り込み, ヒューマン・コンピュータ・インタラクション

8. 通信アーキテクチャ[2]

通信装置, 通信方式, 通信制御, コンピュータネットワーク, ネットワークアーキテクチャ, インタネット, コンピュータ間通信

9. システムプログラムとコンピュータアーキテクチャ[3]

システムプログラムの変遷とアーキテクチャ, プログラミング言語とアーキテクチャ, コンパイラとアーキテクチャ, オペレーティングシステム(OS)とアーキテクチャ, プログラミング言語/コンパイラ/OS/ハードウェアのトレードオフ

先修科目

ハードウェアの構成(コンピュータサイエンス序説)

後修科目

情報ネットワーク, コンパイラ, オペレーティングシステム

関連科目

論理回路, 論理設計, プログラミング言語論

実験及び演習

- アセンブリプログラミング：身近にあるコンピュータ（8ビットマイクロプロセッサやボードコンピュータなど）のマシン語によって簡単なプログラムを作成し，ハードウェア機構について理解する．
- コンピュータの製作実験：関連科目の実験及び演習と統合し（例：CADシステムを用いたLSIの論理設計演習と統合するなど），連続性を持たせる（例：製作したコンピュータ上でシステムプログラムの実験／演習を行なうなど）．「アーキテクチャの設計」を中心にし，実際の製作は共同利用施設である大規模集積システム設計教育研究センター（VDEC）を活用するなどによってオーバーヘッドが生じないようにするのが望ましい．特に，設計だけで終わらずに設計したものを実際にLSIなどにすることが重要である．できるだけ，現代の製作技術に即するのが望ましく，「LSIの設計・製作演習」としたい．

実施方法

- すべての項目を行なうと2半期必要であるが，もし，半期で本科目を実施するならば，1～3の全項目と4～9のうち修得科目として挙げた項目は最低限学習するのが望ましい．

教科書/参考書

- 計算機アーキテクチャ，斉藤 忠夫・笈田 弘・大森 健児，オーム社，1985（教科書）
入出力，通信，OSとのインタフェース，実際のアーキテクチャについての詳解など，必要十分な項目が広く網羅されている．
- 計算機方式，高橋 義造，コロナ社，1985（教科書）
前半はアーキテクチャに関する基礎知識，後半は実際の種々のアーキテクチャの紹介という構成である．
- コンピュータアーキテクチャ 第2版，村岡 洋一，近代科学社，1985（教科書）
マシン命令セット，OS，システム構成，ハードウェア構成などの種々の相異なる機能レベルを切口にアーキテクチャを説明している．
- 計算機システム工学，富田 眞治・村上 和彰，昭晃堂，1988（教科書）
演算装置，メモリ装置，制御装置の各ハードウェア構成方式について詳しい．
- コンピュータの論理構成とアーキテクチャ，楠 菊信・武末 勝・脇村 慶明，コロナ社，1988（教科書）
マイクロプログラム制御方式とメモリ，入出力の各アーキテクチャについて詳述している．ハードウェアとソフトウェアとの関連については「ソフトウェア指向アーキテクチャ」という章を設けている．
- 電子計算機Ⅱ，飯塚 肇，コロナ社，1990（教科書）
演算，命令実行，メモリ，入出力の各機能と制御機構との関連で章立てしてある．
- コンピュータアーキテクチャ原理，曾和 将容，コロナ社，1993（教科書）
計算モデルも含めてノイマン型コンピュータの基本アーキテクチャについて述べ，それをもとに種々の高性能化を図る方式についての説明へ展開している．高性能コンピュータの実例にも言及している．
- コンピュータアーキテクチャの基礎，柴山 潔，近代科学社，1993（教科書）
コンピュータリテラシとしてのアーキテクチャの基礎が丁寧に書いてある．ハードウェアとソフト

ウェアのトレードオフを切口に，アーキテクチャ設計における種々の方式の選択肢が対照して述べられている。

- コンピュータアーキテクチャ，馬場 敬信，オーム社，1994（教科書）
汎用コンピュータと高性能コンピュータのハードウェア構成方式について詳解している。例題が豊富である。
- 計算機アーキテクチャ，橋本 昭洋，昭晃堂，1995（教科書）
演算，命令実行制御，メモリ，入出力の各ハードウェア機構について概説している。OSとの関連では「システムアーキテクチャ」の章を設けてある。
- コンピュータアーキテクチャ，柴山 潔，オーム社，1997（教科書）
マシン命令セット，プロセッサ，メモリ，さらにはマイクロプロセッサやスーパーコンピュータの実例の各機能ごとに，ハードウェアとソフトウェアとのトレードオフの対照を切口に種々の方式について詳解している。また，システムプログラムとハードウェアとの関連についても詳しい。
- コンピュータアーキテクチャI，富田 眞治，丸善，1994（参考書）
演算装置，メモリ装置，制御装置のハードウェア構成について詳解している。やや高度な内容である。また，実際の処理方式として並列処理を取り上げ詳説している。
- コンピュータハードウェア，富田 眞治・中島 浩，昭晃堂，1995（参考書）
ブール代数と論理回路を出発点に演算回路とメモリのハードウェア構成について詳説し，また，簡単な16ビットコンピュータの設計例を与えてある。「論理回路」と「コンピュータアーキテクチャ」との関連やハードウェア実験についての参考書となる。
- ハードウェア入門，柴山 潔，サイエンス社，1998（参考書）
コンピュータのハードウェア構成について平易に書かれ，読み解くのに専門的な知識を必要としない入門書である。したがって，コンピュータリテラシ教育用の教科書に最適であるが，アーキテクチャの観点からは書かれていないので，本講義項目では参考書として挙げる。
- 計算機アーキテクチャと構成方式，中澤 喜三郎，朝倉書店，1995（教員用参考書）
論理素子や実装技術から最近の高性能コンピュータまでのアーキテクチャに関連する項目のほとんどを網羅・詳解してある。コンピュータアーキテクチャに関する詳細なハンドブックとして教員向けの参考書となる。
- コンピュータの構成と設計 - ハードウェアとソフトウェアのインタフェース - ，D.A.Patterson, J.L.Hennessy（成田 光彰 訳），日経 BP 社，1996（参考書，教員用参考書）
豊富な例題をもとに，コンピュータの性能向上のポイントを実際のかつ定量的に詳述している。原書の第2版が1997年に出版されている。

備考（コメント）

- コンピュータリテラシ教育や「コンピュータサイエンス序説」で行なうべき「ハードウェアの構成」についての講述ばかりに偏らないように注意する。
- 「コンピュータサイエンス序説」中の「ハードウェアの構成」とは別科目にするのが望ましいが，実現可能性などの点でそれが困難な場合には，本科目の一部を「序説」に回しても良い。

U-5 プログラミング言語論 [1]

目的

プログラミング言語において構文と意味があることと、それらがそれぞれの言語においてどのように定義されているかを理解させる。データ構造やアルゴリズムの表現法、および、構造化や抽象化などのプログラミングにおいて重要な概念が、それぞれの言語でどのように定義されているかを示して、種々の言語に共通する概念や機能と、それぞれの言語を特徴付ける概念や機能とがあることを理解させて、プログラミング言語を客観的な対象物として取り扱う態度を身に付けさせる。

概要

プログラムを構成するために有効な構造化や抽象化の概念を理解し、それらを記述する方法を習得する。命令型、関数型、論理型等の種々のプログラミング言語について、それぞれの特徴と主要な機能を説明する。

[修得項目]

構文、意味、データ構造、制御構造、モジュール構造、データ抽象化、型理論、型検査、構造化と抽象化

[知識項目]

構文規則、具体構文と抽象構文、帰納的定義、構造帰納法、型推論、部品化、再利用、並行動作

[関連項目]

計算モデル、意味領域

講義項目

1. プログラミング言語の歴史 [1]

抽象化、構造化、プログラミングパラダイム

2. 構文と意味 [2]

構文図式、BNF、抽象構文、静的意味、動的意味、計算モデル、形式的意味論

3. 手続き型言語 [3]

制御構造、データ構造、モジュール構造

4. 関数型言語 [3]

型理論、型検査、型推論、高階関数、リスト構造

5. 論理型言語 [2]

関係、述語、単一化

6. オブジェクト指向言語 [2]

モデル化、オブジェクト、クラス、部品化、再利用

7. 並行動作プログラム [2]

並行動作プロセス、同期、通信、safety、liveness

先修科目

プログラミング入門，ハードウェアの構成(コンピュータサイエンス序説)，離散数学，データ構造とアルゴリズム，形式言語とオートマトン

後修科目

コンパイラ，ソフトウェア工学

関連科目

コンピュータアーキテクチャ，計算論，プログラム意味論

実験及び演習

- プログラミング言語の設計演習：
例えば，while プログラムにいくつかの制御構造を追加させて，その構文と意味を記述させる．時間があれば，実現も行なわせる．
- ユーザコマンドの設計演習：
比較的単純なシステム(簡単なデータベース，エディタ，制御システム等)に対するユーザコマンドを設計させて，その構文と意味を記述させる．時間の余裕があれば，実現も行なわせる．

実施方法

- 手続き型言語と，関数型言語または論理型言語との，二種類以上のプログラミング言語についての知識ないし利用経験があれば，望ましい．
- 種々の言語に関する言語仕様の表面的な紹介にならないように留意すること．
ML などの関数型言語や Pascal あたりで，プログラミングにおける基本的で重要な概念や機能に習熟させた後に，異なる種類の言語の間で共通する概念や機能があることと，それぞれに特有の概念や機能があることを理解させると効果的かもしれない．

教科書/参考書

- 武市正人：プログラミング言語，岩波書店，1994. (教科書)
本科目の目的や内容にほぼ合致した本である．
- R. D. Tennent : Principles of Programming Languages, Prentice-Hall, 1981.
竹下亨 訳：プログラミング言語の原理，日本コンピュータ協会，1985. (教科書)
内容はやや古くなっているが，定評のある良い教科書である．
- 山崎利治：プログラミング言語，昭晃堂，1989. (参考書)
命令型の汎用言語に限ってはいるが，含蓄ある内容を持つ本である．
- Ravi Sethi : Programming Languages – Concepts & Constructs –, 2nd Edition, Addison-Wesley, 1996. (参考書)
広範な内容を具体的な記述で見通しよくまとめた良書．旧版は翻訳あり． 神林靖 訳：プログラミング言語の概念と構造，アジソン・ウェスレイ・パブリッシャーズ・ジャパン，1995. 1995.

U-6 論理設計 [1]

目的

CMOS を前提とした効率の良い論理回路の設計手法を理解させる。組合せ回路の最適設計手法，順序回路の最適設計手法などを理解させ，論理合成の原理を把握させる。また，ハードウェア記述言語による論理設計にも触れさせる。

概要

デジタル回路の基本となる CMOS 論理回路の設計手法を理解し，効率の良い論理回路が設計できる能力を身につけることを目指す。与えられた論理関数に対して，それを実現する最適な組み合わせ回路を実現する手法を修得する。さらに，与えられた順序機械の仕様に対して，状態数の最小化，状態割り当て，論理設計の各過程を理解し，同期式順序回路による順序機械の実現手法を修得する。ハードウェア記述言語による論理設計の原理を理解させ，ソフトウェアと同程度の作業でハードウェアを設計できることを教える。

[修得項目]

基本素子の構造と機能，組み合わせ回路の設計法，同期式順序回路の設計法

[知識項目]

組合せ回路の最適化，ドントケアの取扱い，同期式と非同期式の違い，制御部とデータパスの分離

[関連項目]

論理設計用 CAD ツールとその原理

講義項目

1. 基本論理素子の回路 [2]

基本ゲートの回路構造と動作原理，記憶素子の回路構造と動作原理，各種メモリの回路構造

2. 組み合わせ論理回路の最適化設計 [4]

2 段論理最小化（カルノー図などによる），多段論理最小化，テクノロジーマッピング，回路の評価尺度と評価方法，ドントケアの扱い，ROM/PLA などによる実現

3. 同期式順序回路の最適化設計 [3]

状態数最小化，状態割当，完全指定と不完全指定順序回路

4. 非同期式順序回路の設計 [2]

同期式と非同期式の差異，非同期式の利点，ハザードと競合

5. 制御系とデータパス [2]

システムの設計手法，制御部とデータパスの分離の意義，データパス回路の設計手法

6. ハードウェア記述言語による論理設計 [2]

ハードウェア記述言語の概要，ハードウェア記述言語の文法，言語と合成ツールの関係，合成ツールの原理，シミュレーションによる設計検証

先修科目

論理回路，形式言語とオートマトン，

後修科目

集積回路工学

関連科目

コンピュータアーキテクチャ，データ構造とアルゴリズム，信号処理，数理論理学

実験及び演習

- 論理回路の製作実験：簡単な順序回路（簡単な演算回路など）を試作させる．SSIなどで試作する方法，FPGAなどのプログラマブルデバイスを利用する方法，CADツール上で設計だけを体験させる方法など，実施方法は種々可能である．ハードウェア記述言語の所理系など設計用のツールは東京大学の大規模集積システム設計教育研究センター（VDEC）を利用して安価に大量に提供を受けることが可能である．

教科書/参考書

- 論理回路理論，山田輝彦，基礎情報工学シリーズ，森北出版，1990（教科書）
論理回路の性質から合成法までわかりやすく説明されている．
- 論理回路，高木直史，New Text 電子情報系シリーズ，昭晃堂，1997（教科書）
古典的な論理回路に関する事項をコンパクトにまとめて平易に解説している．
- 論理設計スイッチング回路理論，笹尾勤，近代科学社，1995.（教科書）
論理関数の性質について詳しい．理論的に良く整理された教科書である．
- Switching and Finite Automata Theory, Second Edition, Zvi Kohavi, McGraw Hill, 1978.（教科書）
明快な英語と整理された記述の論理関数や論理回路に関する古典的教科書．英語の勉強にもよい．
- 論理回路の基礎，田丸啓吉，工学図書，1989.（教科書）
論理回路設計の基礎を平易に解説している．
- 情報工学実験，池田克夫 編，新コンピュータサイエンス講座，オーム社，1993（実験の教科書）
ソフトウェアから論理回路まで一貫した教科書．
- Introduction to Digital Logic Design, J.P. Hayes, Addison Wesley, 1993.（教科書）
- Digital Design -Principles and Practice - 2nd. Ed., J. Wakerly, Prentice Hall, 1994.（教科書）
- Principles of Digital Design, Daniel Gajski, Prentice Hall, 1997.（教科書）

U-7 オペレーティングシステム [1]

目的

本科目の狙いは、以下である。

- オペレーティングシステム (OS) の役割、基本的概念および実現方式を理解させること。
- 既存の小さなオペレーティングシステムを変更できる能力を養成すること。

概要

OS をハードウェアとユーザとの間に位置するプログラムという立場から、OS が果たすべき役割、目的、OS 機能の概念および実現方式について講述する。

[修得項目]

(大項目) OS の役割と目的および基本機能、仮想化、ユーザインタフェース、効率

(小項目) システムコール、グラフィカルユーザインタフェース、多重プログラミング、プロセス、コンテキストスイッチング、スケジューリング方式、割り込み、仮想メモリ、ページング方式、セグメンテーション方式、ファイル構造、ディスク構成、ディスクスケジューリング、並行プロセス、同期、排他制御、クリティカルセクション、不可分命令、セマフォ、デッドロック、保護

[知識項目]

OS の変遷、OS 構成法 (基本的分類 (モジュール化、階層化)、マイクロカーネルアーキテクチャなど)、ポリシとメカニズムの分離、軽いプロセス (スレッド)、OS のインストール

[関連項目]

分散/並列処理、リアルタイム処理、マルチメディア処理、QoS(Quality of Service)

講義項目

1. OS の基本概念 [1]

OS の歴史および変遷、ソフトウェアの構造、コンピュータハードウェアの仮想化、システムコール、ハードウェアの基本知識 (CPUアーキテクチャ、命令セット、アドレス空間、割り込み)

2. ユーザからみた OS[1]

コマンドの実行、グラフィカルユーザインタフェース (ウインドウシステム、アイコン、メニュー、ポインティングデバイス)、日本語処理

3. OS の役割と機能 [1]

プロセス/メモリ/ファイル/入出力/ネットワーク管理、サービスの種類

4. プロセス [1]

プロセスの定義、軽いプロセス、多重プログラミング、コンテキストスイッチング、プロセス制御ブロック、スケジューリング、レディキュー

5. 並行プロセス [1]

記述方法，同期，排他制御，クリティカルセクションの定義とその解決策（ソフトウェア，ハードウェア），不可分命令，セマフォ，古典的なプロセス協調問題

6. デッドロック [1]

デッドロックの必要条件（検出），防止，回避，回復

7. メモリ管理 [2]

スワッピング，2次記憶，多重区画割当，アドレス空間，保護，ページング，セグメンテーション，内部/外部断片化，仮想記憶，デマンドページング，ページ置き換えアルゴリズム，ワーキングセット，実現方法

8. ファイルシステムと補助記憶 [1]

ファイルの概念，名前管理，ディレクトリ/ファイル構造，ディスク装置の構造，割当/アクセス技法，ディスクスケジューリング

9. 入出力管理 [1]

入出力ハードウェアとその制御，割り込み

10. 保護とセキュリティ [1]

アクセス制御リスト，ケイパビリティ，認証

11. OS 設計法 [1]

OS アーキテクチャ，OS 構成法（モジュール化や階層化による基本的分類，単層カーネル，マイクロカーネル），ポリシとメカニズムの分離，ヒューマンインタフェース

12. OS 事例 [1]

過去の OS（THE, Multics, OS/360, など），UNIX 系，Windows

13. 先進的 OS 概論 [1]

分散/並列システム，ネットワーク，マルチメディア，リアルタイム，QoS(Quality of Service)

14. OS のインストール概論 [1]

インストール手順，インストール支援ツール，ネットワークハードウェア環境，カーネルのカスタマイズ

先修科目

コンピュータサイエンスリテラシ（コンピュータサイエンス序説，プログラミング入門），データ構造とアルゴリズム，コンピュータアーキテクチャ，コンパイラ，プログラミング言語論

後修科目

情報ネットワーク，データベース

関連科目

ソフトウェア工学，ヒューマン-コンピュータ・インタラクション

実験及び演習

- 簡単な OS の作成：全ての管理機構を含んだ OS の作成は，各種事情（時間，人材，経費）から困難なので，問題を絞った OS を構築させる．例えば，スケジューリング，特定の応用プログラムの高速処理，など．
- 既存 OS のインストールと変更：既存 OS(例えば，UNIX 系) をインストールし，さらに，その一部(例えば，スケジューリング) を変更してみる．
- ネットワーク対応のコンピュータシステムの構築：既存の単体のコンピュータシステムをネットワーク(例えば，イーサネット) に有機的に接続させる．このとき，複数のネットワークボードを用いて，ネットワークボードの違いによって，インストールがどのように異なるかを体験する．

実施方法

- 上記は，半期(半年)を想定しており，多くの大学で実施可能と思われる．

教科書/参考書

- オペレーティング・システムの機能と構成，高橋 延匡・土居 範久・益田 隆司，岩波書店，1983(参考書)．
少し古いが，著者はメインフレーム上の OS 開発者であり，その立場から，単なる技術の記述のどまらず，OS の歴史，および開発思想など，その背景にある考え方を記述しているところに特徴がある．
- オペレーティングシステムの概念(上)，宇津宮 孝一・福田 晃 共訳，培風館，1987(参考書)(原著: Operating System Concepts, 2nd edition, J.L.Peterson and A.Silberschatz, Addison-Wesley, 1985)．
原著は教科書としてアメリカでよく使用されている．OS の基本概念を論じている．クリティカルセクション問題のソフトウェアによる解決アルゴリズムが記載されているところに特徴がある．邦訳は，第 2 版であるが，原著ではすでに，第 3 版(1991)，第 4 版(1994) が出版されており，分散システムなどの最近の話題が追加されている．
- オペレーティングシステム，前川 守，岩波書店，1988(参考書)．
OS の基本概念および機構についてかなり詳しく述べている．また演習問題も豊富である．
- オペレーティングシステム，清水 謙多郎，岩波書店，1992(参考書)．
一般的な概念の記述だけでなく，最近の技術動向も述べている．
- オペレーティングシステム，谷口 秀夫，昭晃堂，1995(参考書)．
OS の「概念」および「構造」と，ユーザの「利用」との関係など，丁寧に記述している．
- UNIX カーネルの設計，坂本文・多田好克・村井純 共訳，bit 別冊，共立出版，1990(教員用参考書)．(原著: The Design of the UNIX Operating System, M.Bach, Prentice Hall, 1986)．
UNIX System V 系について，詳述している．
- Modern Operating Systems, A.S.Tanenbaum, Prentice-Hall, 1992(教員用参考書)．
古典的な OS から近代の OS(主に分散 OS)までを網羅している．

- The Design and Implementation of the 4.4BSD Operating System, M.McKusick, K.Bostic, M.Karels, and J.Quarterman, Addison-Wesley, 1996 (教員用参考書) .
新しいUNIX BSD 系 OS である , 4.4BSD について , 内部構造などを詳細に記述している .

備考 (コメント)

- 最近の動向である , グラフィカルユーザインタフェースなどのユーザと OS とのインタフェースはぜひとも , 修得させたい .

U-8 コンパイラ [1]

目的

本科目では、プログラミング言語のもっている言語モデルと、OS / コンピュータアーキテクチャの間を目的に応じて埋めるというコンパイラの役割と構成および実現方法を理解させるとともに、簡単なコンパイラを作成できる基礎的能力を養成する。

概要

本科目では、プログラミング言語とOS / コンピュータアーキテクチャの間のギャップを埋めるコンパイラの役割、構成、および実現方法などについて、小規模な言語仕様のプログラミング言語と簡潔なコンピュータアーキテクチャを題材にとりあげながら講述する。

[修得項目]

コンパイラの目的と構成(字句解析, 記号表, 構文解析, 意味解析, 中間コード生成, 最適化, コード生成), コンパイラとインタプリタ, 原始プログラムと目的プログラム, 字句解析の構成, 正規表現, 字句解析器(構成, 文字読みとり, 字句読みとり), BNF 記号, 文脈自由文法, 解析木, 下向き構文解析法, 上向き構文解析法, 意味解析の役割と位置づけ, スコープ, スコープ規則, 型検査, 目的コード生成方法(式, 文の列)

[知識項目]

フロントエンドとバックエンド, パス/フェーズ, 記号表, 中間コードの種類(構文木, 非循環有効グラフ, 後置コード, 4つ組, 3つ組), 最適化例(コピー伝搬, 共通部分式の除去, 不要コードの除去, インライン展開など), 基本ブロック, 制御フロー解析, データフロー解析, ループの最適化

[関連項目]

リンカ/ローダの役割と位置づけ, 有限オートマトン, 属性文法, メモリ管理, スタック, ヒープ, 動的メモリ管理, ごみ集め, 字句解析器生成系(例: lex), 構文解析器生成系(例: yacc), 意味解析器生成系, コード生成器生成系

講義項目

1. プログラム言語と言語処理系 [1]

コンパイラの目的と歴史, プログラム言語のモデル/パラダイムとコンパイラの役割, コンパイラとインタプリタ, 原始プログラムと目的プログラム, 評価尺度

2. BNF によるプログラミング言語の定式化 [1]

BNF, 対象の規則化とその表現, 有限オートマトン

3. コンパイラの機能と構成 [1]

字句解析, 記号表, 構文解析, 意味解析, 中間コード生成, 最適化, コード生成, リンカ/ローダの役割と位置づけ, パス/フェーズ, フロントエンドとバックエンド

4. 字句解析 [2]

字句解析の構成, 正規表現, 有限オートマトン, 字句解析器(構成, 文字読みとり, 字句読みとり), 字句解析器生成系(例: lex)

5. 構文解析 [2]

文脈自由文法，解析木，下向き構文解析法（LL 構文解析など），上向き構文解析法（演算子順位構文解析，LR 構文解析など），構文解析器生成系（例：yacc）

6. 意味解析 [2]

役割と位置づけ，スコープ，型検査，属性文法，スコープ規則，記号表，意味解析器生成系

7. 実行時環境 [1]

メモリ管理，スタック，ヒープ，引数の渡し方，動的メモリ管理，ごみ集め

8. 中間コード生成 [2]

中間コードの種類（構文木，非循環有効グラフ，後置コード，4つ組，3つ組），代入文と配列の中間コード，制御文と論理式の中間コード，手続き呼び出しの中間コード

9. コード生成 [1]

中間コードと目的コード，式のコード生成，文の列に対するコード生成，コード生成器生成系

10. コード最適化 [1]

最適化例（コピー伝搬，共通部分式の除去，不要コードの除去，インライン展開など），基本ブロック，制御フロー解析，データフロー解析，ループの最適化

11. 事例研究 [1]

いくつかのコンパイラ/インタプリタの事例を紹介する（例；Javaのインタプリタ）。

先修科目

プログラミング入門，データ構造とアルゴリズム，形式言語とオートマトン，コンピュータアーキテクチャ，プログラミング言語論

後修科目

とくにはない。

関連科目

オペレーティングシステム，ソフトウェア工学

実験及び演習

- コンパイラの実際：小規模な言語仕様のプログラミング言語（tiny Basic, PL/0, tiny Pascal, C-）のコンパイラを読み，拡張する。
- 自分だけのプログラミング言語とコンパイラ的设计と实现：身の回りにある事象（何でもよい。高性能電卓でも，ゲームでも，論理回路でも）を数個～数十個程度のBNFで記述し，その言語Aからある出力Bへ変換するプログラムを作る。

実施方法

- 上記は、半期(半年)を想定しており、多くの大学で実施可能と思われる。

教科書/参考書

- 中田育男：コンパイラ，コンピュータサイエンス・ライブラリ，産業図書，1981(教科書)。
若干古いが，現在でも十分教科書として利用可能な良著．著者はメインフレーム上の FORTRAN コンパイラの開発者であり，その立場から，各種解析技術の理論面から，コンパイラ開発の実際面まで幅広い知識を与えてくれる．
- 中田育男：コンパイラ，オーム社，1995(教科書)。
ソース言語を主に PL/0 に絞り込んだ上で，コンパイラ全体を一貫して，分かり易く記述されている．その意味で，PL/0 類似の言語用のコンパイラ開発を取り上げる実習などでの教科書あるいは参考書として最適であろう．
- 土居範久 訳：コンパイラ，情報処理シリーズ 7，培風館，1986(教科書)。
(原著，Alfred V.Aho and Jeffrey D.Ullman：Principles of Compiler Design, Addison-Wesley, 1977)
幅広く，かつ深くこの分野を一覧できる良書．参考文献も豊富で，参考書としても好適．逆に学部での入門教育にはやや詳細すぎるくらいもある．通年の講義であれば教科書としても使用できよう．
- 原田賢一 訳：コンパイラ 原理・技法・ツール(全二巻)，サイエンス社，1990(教科書)。
(原著，Alfred V.Aho, Ravi Sethi, and Jeffrey D.Ullman：Compilers Principles, Techniques, and Tools, Addison-Wesley, 1986)
著者らは上記前書が「コンパイラ設計の入門コースでの教科書」(本書まえがきより)と位置付けているのに対し，本書は重要な話題についてさらに深く掘り下げた集大成的専門書．
- 佐々政孝：プログラミング言語処理系，岩波講座 ソフトウェア科学 5，岩波書店，1989(参考書)。
この分野の一般的な基本概念の解説と共に，属性文法も詳解するなど，どちらかという理論面に重点をおいている．しかも，PL/0 言語用の字句/構文解析部について，実際に lex/yacc を使用する記述例も記載．理論的詳細まで理解できるので，大学院レベルでの教科書に好適．
- 今泉貴史 訳：コンパイラ設計技法 理論と実践，トッパン，1993(参考書)。
(原著，Thomas Pittman and James Peters：The Art of Compiler Design *Theory and Practice*, Prentice Hall, 1992)
上と同じく，属性文法，特に transformational attribute grammar に焦点を当て詳解．同時に，それを利用したコンパイラ作成の自動化など最近の技術動向も豊富な実例と共に解説．また，例えば，難解な部分を示す印を付すなどし，教科書としての利用を目指し読者の理解を助ける工夫がいくつかなされている．半期の講義の教科書とするには分量が多い．
- 辻野嘉宏：コンパイラ，情報工学入門選書 10，昭晃堂，1996(参考書)。
C 言語のサブセットを対象としたサンプルコンパイラのソースコードが，付録に記載されている．

備考(コメント)

- 小規模な言語仕様でよいからプログラミング言語を作成するということを学生に意識させた講義がよい．

U-9 デジタル通信 [1]

目的

現在，コンピュータ技術はネットワーク技術と結び付くことにより，地理的な制約のない利用が可能になっている．本講義では，デジタル通信の基本原理を知るとともに，コンピュータネットワークにおいてデジタル通信技術がどのように活用されているかを教える．現在，コンピュータネットワークはコンピュータ科学者・技術者にとって必要不可欠になっている．そこで「ネットワークリテラシ」として早い段階で教えることも考えられる．その場合には，以下の項目 1～3，9～11 が講義の対象になる．その時は，それぞれ個々の技術の詳細よりも，通信の技術課題がどのように解決されてきたか？ どうしてそれが必要だったのか？ 何が残された問題か？ が重要で，それらが修得すべき項目となる．また，例えば，符号理論についても理論を教えること自体が目的ではなく，通信の分野でどのように役立てるかが重要で，後修科目の「符号理論」に対する動機付けが目的である．従って「イーサネットの CRC フィールドの長さは何バイトになっているか？ なぜ，そう決められたのか？」を教えるべきである．

概要

現在，コンピュータ技術はネットワーク技術と結び付くことにより，地理的制約のない利用が可能になってきている．本講義では，デジタル通信の基本原理を知るとともに，コンピュータネットワークシステムにおいてそれがどのように活用されているかを学ぶ．

[修得項目]

データ通信システムの構成，通信回線，モデム，パケット交換と回線交換，電話回線，ISDN 回線，専用線，LAN(イーサネット)，無線 LAN，移動体通信，パーソナル通信システム(携帯電話、PHS など)，衛星通信，インターネット

[知識項目]

通信路符号化定理，誤り検出・訂正，2進符号，基本型データ伝送制御手順，HDLC，振幅変調，周波数変調，位相変調，TDMA，FDMA，パケット多重，WDM，静止衛星，低軌道衛星

[関連項目]

基本波形と伝送路，伝送路ひずみ，等化器

講義項目

1. データ通信の歴史 [2]

電話網からデジタル通信網，計算機ネットワーク，無線システムへの発展

2. データ通信システムの構成 [1]

ハードウェア構成 / ソフトウェア構成；なぜモデムが必要なのか？

3. データ通信システムの構成：通信回線 [2]

パケット交換と回線交換．電話回線，ISDN 回線，LAN，専用線，電子メールはなぜ相手に届くのか？，インターネットでホームページが見られるのはどうしてか？

4. 符号理論 [2]

通信路符号化定理，誤り検出・訂正

5. データ伝送制御手順 [2]

2進符号，基本型データ伝送制御手順，HDLC

6. 基底帯域伝送 [2]

基本波形と伝送路，伝送路ひずみ，等化器

7. 変復調方式 [2]

振幅変調，周波数変調，位相変調

8. 多重化方式 [1]

TDMA，FDMA，パケット多重，WDM

9. 無線系 [2]

無線 LAN，移動体通信，PHS と携帯電話はどこが違うのか？，ノート型パソコンでインターネットを利用するしくみ

10. 衛星通信 [1]

衛星の種類，TDMA

11. データ通信の今後 [2]

現状 (N-ISDN/B-ISDN) と今後の発展，光通信のゆくえ

12. 情報化社会と倫理 [1]

情報化社会の発展に伴って顕在化しつつある倫理問題（いわゆるネチケットに触れる）

先修科目

確率論と情報理論，信号処理（ネットワーク入門科目とする場合，これらは必須ではなく，むしろ後修科目としてそれらの科目の動機付けとなるようにすべきである）

後修科目

情報代数と符号理論，情報ネットワーク

関連科目

コンピュータアーキテクチャ

実験及び演習

特になくてよいが，ネットワーク入門とする場合はインターネットの利用（インターネットに触れてみる）が考えられる。また，コンピュータリテラシ教育が終わっていれば，WWW のホームページ作成なども考えられよう。

実施方法

- デジタル通信の科目とする場合は項目 1～8。また、項目 1～3, 9～11 によって「ネットワーク入門」とする。その場合にはインターネットを中心にして講義や演習を行なうことも考えられる。時期はコンピュータリテラシ科目の直後に行なっておくことが望ましい。

教科書/参考書

- Digital Communications with Fiber Optics and Satellite Applications, Harold B. Killen, Prentice-Hall, 1988 (教科書)(邦訳「デジタル通信技術の基礎と応用」情報通信システム研究会編, 企画センター, 1991)
デジタル通信技術に関する入門書
- デジタル通信, 木村英俊, 広崎膨太郎, 丸善, 1991 (教科書)
デジタル通信技術に関する入門書
- 新情報通信早わかり講座, 井上伸雄, 都丸敬介, 日経 BP 社, 1995 (参考書)
最近の通信に関する技術動向の平易な解説書
- 通信サービス利用ガイドブック, 日経 BP 社, 1996 (教員用参考書)
通信サービスに関する最新情報が網羅されている

U-10 データベース [1]

目的

関係データベースを基本に置き，データモデルとデータベースシステムの基本概念を学習させ，データ設計やデータ操作，データ管理の原理と方法を理解させる．

概要

大量のデータを効率良くコンピュータで処理するためには，それらをデータベースとして管理することが重要である．本科目では，関係データベースを主たる対象にして，データベース管理システムの基本概念と，基礎的な知識を学習する．

[修得項目]

データベース管理システムの基本的概念，データモデル，関係データモデル，データ定義とデータ操作，同時実行制御と障害回復，分散データベース

[知識項目]

3層スキーマ，一貫性制御，関係設計と正規化，関係代数と関係論理，問合せ最適化，SQL，トランザクション，2相施錠，2相コミット

[関連項目]

関係と論理，種々のデータモデル（演繹データベース，オブジェクト指向データベース，意味データモデル，ERモデル），データベースマシン，情報検索，バンキングシステム，データウェアハウス，データマイニング，リアルタイムデータベース，マルチメディアデータベース

講義項目

1. データベースシステムの基礎概念 [2]

データ共有，一貫性制御，データ独立性，データモデル，3層スキーマ，データ辞書

2. データモデル [2]

階層モデル，網モデル，実体関連モデル，その他モデル

3. 関係モデル [1]

データ構造，諸概念，キー

4. 関係操作 [2]

関係代数，関係論理

5. 関係設計論 [1]

従属性制約，非正規関係，正規形，正規化

6. 関係データベース言語 SQL[1]

データ定義，問合せ，更新，ビュー

7. 物理的格納方式 [2]

レコード，ヒープファイル，ハッシュファイル，索引付きファイル，B(+)木，

8. 問合せ処理 [1]

実行プラン，最適化，コスト計算

9. 同時実行制御 [1]

トランザクション，直列可能性，施錠/解錠，2相施錠，同時実行制御の諸方式

10. 障害回復 [1]

障害の種類，ログによる回復，チェックポイント，メディア障害

11. 分散データベース [1]

分散の問題，位置透明性，2相コミット，マルチデータベース，連邦データベース

12. 実システム [1]

先修科目

データ構造とアルゴリズム，オペレーティングシステム，数理論理学，プログラミング言語論

後修科目

ヒューマン-コンピュータ・インタラクション

関連科目

ソフトウェア工学

実験及び演習

- データベース演習：実際にデータベースを作り，簡単な応用プログラムを書いてみる。

教科書/参考書

- データベースシステム，北川博之，昭晃堂，1996。（教科書）
理論的な面から実装にいたるまでバランス良くまとめられており，この科目の目標に適した教科書である。

U-11 人工知能 [2]

目的

人工知能は常に CS の夢の部分を担い続けてきた。CS の黎明期においてはフォンノイマンが人間とコンピュータの関係を論じ、シャノンがゲームの探索戦略を提案している。今日においては、知能ロボットやサイバースペースでの疑似生命とその進化が議論されている。

しかしながらその一方で、人工知能技術はコンピュータの処理能力を敏感に反映しながら進歩してきた。まず計算能力の拡大に伴って状態空間探索が、次に記憶空間の拡大に伴って知識表現が、さらにネットワークの発達に伴って分散人工知能が、そして今日では膨大なデータと計算を前提とした機械学習が研究されている。

人工知能は人間の知能をモデル化し、コンピュータシステム上に実現することを目標とした尽きることのない分野である。その間に生み出された技術蓄積を体得させることが講義の目的であるが、技術のみの講義に終始したのではこの分野を十分に語り尽くせない。人工知能研究の歴史を、その成功事例も失敗事例も含めて、コンピュータの歴史的状況と合わせて説明することが必要である。講義を通じて、CS の面白さと可能性を感じさせることも講義のもう一つの目的である。

概要

人間と同等の知能をコンピュータ上に実現することを目標とした人工知能研究の歴史と到達点を述べる。また、基本的な計算と表現のモデルである状態空間探索と論理についてその基礎を講義する。さらに、それらのモデルを用いた問題解決、知識表現、機械学習などの様々な技術を講義する。さらに複数の人間の相互作用のモデル化を目標とする分散人工知能に触れる。最後に人工知能のこれまでの応用と今後の展望を紹介する。

[修得項目]

状態空間、経路計画、深さ優先探索、幅優先探索、最良優先探索、A*アルゴリズム、制約充足、弛緩法、2人ゲーム、ミニマックス手続き、手続き、命題論理、第1階述語論理、導出原理、論理プログラミング、プロダクションシステム、問題解決、プランニング、探索制御、非単調論理、継承、意味ネットワーク、フレーム表現、演繹学習、機能学習、神経回路網、逆誤差伝播法、遺伝アルゴリズム、分散問題解決、交渉、均衡

[知識項目]

人工知能の歴史、コンピュータチェス、エキスパートシステム、フレーム問題、大規模知識ベース、Prolog、Lisp、関数型言語、オブジェクト指向言語、シミュレーテッドアニーリング、クラシファイアーシステム、エージェント、自己組織化

[関連項目]

知能ロボット、ヒューマンインタフェース、自然言語理解、画像理解、音声認識、人工生命、複雑系、拡張現実感、仮想現実感、感性情報処理

講義項目

1. 導入 [2]

人工知能研究の歴史、到達点

2. 状態空間探索 [4]

状態空間モデル、経路計画、制約充足、2人ゲーム

3. 論理と証明 [4]

命題論理, 一階述語論理, 定理証明, 論理プログラミング

4. 問題解決 [4]

プロダクションシステム, プランニング, 問題解決過程の制御

5. 知識表現 [4]

非単調論理, 意味ネットワーク, フレーム

6. 機械学習 [4]

演繹学習, 帰納学習, 神経回路網, 遺伝アルゴリズム

7. 分散人工知能 [4]

合理的エージェント, 分散問題解決, 交渉, 均衡, 組織の形成

8. 応用 [4]

コンピュータチェス, エキスパートシステム, 知能ロボット, ネットワークエージェント, 社会的相互作用

先修科目

形式言語とオートマトン, 数理論理学, データ構造とアルゴリズム, 数理計画法

後修科目

情報ネットワーク, データベース, ヒューマン-コンピュータ・インタラクション

関連科目

画像情報処理, パターン認識, 自然言語処理,

実験及び演習

- 問題解決システムの製作実験: 課題を選択させ, コンピュータを用いてシステムを開発させる. (例: オセロプログラム, 自律移動ロボットのシミュレーション, 電子秘書システム)

実施方法

- 2 半期必要であるが, 半期で実施するならば, 1~5 を中心に学習するのが望ましい.

教科書/参考書

- N. Nilsson (白井他 訳), 人工知能の原理, 日本コンピュータ協会, 1983 (上級者向き教科書) 問題解決を中心にした教科書である.
- Winston, *Artificial Intelligence*, Addison Wesley, 1992 (教科書) 機械学習に関する記述に優れた教科書である.

- Matt Ginsberg, *Essentials of Artificial Intelligence*, Morgan Kaufmann Publishers, 1993 (教科書)
技術指向でバランスのとれた教科書である。
- Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1995 (教科書)
エージェントの視点から人工知能を述べた教科書である。
- A. Barr and E. A. Feigenbaum (田中他監訳), 人工知能ハンドブック, 共立出版株式会社, 1983 (教員用参考書)
この分野を幅広く知るのに良い。
- 白井良明, 人工知能の理論, コロナ社, 1992 (教科書)
基本的な理論を中心とした教科書である。
- ルーメルハート, 人間の情報処理, サイエンス社, 1979 (参考書)
神経回路網を基礎とした教科書である。
- C. Chang and R. C. Lee, *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, Inc., 1973 (参考書)
論理に関する記述が丁寧である。
- 長尾 真, 淵 一博, 論理と意味, 岩波講座 情報科学, 7, 1983 (参考書)
論理とその自然言語への応用の流れを知るのに良い。
- 原尾 政輝, 人工知能アルゴリズム論, 近代科学社, 1993 (参考書)
アルゴリズムの観点から書かれた人工知能論である。
- 石田, 片桐, 桑原, 分散人工知能, オーム社, 1997 (参考書)
分散人工知能関係の研究の流れを知るのに良い。

U-12 情報ネットワーク [1]

目的

情報ネットワークを設計し，構築する上で基礎となる階層化アーキテクチャの概念を知り，ネットワークを介して情報がどのように伝送，処理されて相手に伝えられるのか，さらに情報通信機能を用いてどのようなサービスが実現されているのかを会得させる．特に講義を通して，ネットワーク性能とコスト間に存在するトレードオフ関係を理解させる．

そのためには，各講義項目における事例紹介が重要である．事例紹介では，それぞれの項目に関して技術課題がどのように解決されたか？ どうしてそれが必要だったのか？ 何が残された問題か？ 最新の動向は？などを中心に紹介すべきである．それぞれの時代に最善とされた技術が，関連技術の進展によって最善でなくなる例は数多い．それらの説明を重点的に行なう必要がある．最新のネットワーク技術のすべてを網羅しようとすれば表面的な技術動向の紹介に終始しがちであるが，なぜ対立する概念（例えば，パケット交換と回線交換，コネクション型とコネクションレス型通信など）が現実のネットワーク技術として存在するかを教えることが肝要である．

概要

情報ネットワークを設計し，構築する上で基礎となる階層化アーキテクチャの概念について述べ，ネットワークを介して情報がどのように伝送，処理されて相手に伝えられるのか，さらに情報通信機能を用いてどのようなサービスが実現されているのかを知る．特に講義を通して学ぶべきことは，ネットワーク性能とコスト間に存在するトレードオフ関係を理解することである．

各講義項目では，最新の技術動向も紹介する．特に，現実のネットワークでは，対立する概念（例えば，パケット交換と回線交換，コネクション型通信とコネクションレス型通信など）が実現されており，それらを順次紹介していくことにより，なぜ，対立した概念に基づくネットワーク技術が実際に存在し得るのかを明らかにしていく．

[修得項目]

プロトコルの階層化，ベースバンド伝送，誤り制御，フロー制御（ウィンドウフロー制御，レート制御），メディアアクセスプロトコル（アロハ方式，CSMA/CD方式，トークンパッシング方式など），パケット交換と回線交換，コネクション型／コネクションレス型通信，輻輳制御，経路制御

[知識項目]

TCP/IP，OSI，伝送媒体（より対線，同軸ケーブル，光ファイバ，無線，衛星回線），イーサネット，マルチメディア通信，SNMP，ARP，DNS，クライアント・サーバモデル，ファイル転送プロトコル／遠隔ターミナル機能／電子メール／WWWなどのネットワークプロトコルやサービス機能

[関連項目]

オペレーティングシステム，コンピュータアーキテクチャ，画像／音声の符号化圧縮，マルチメディア処理

講義項目

1. 情報ネットワークとは [1]

広域ネットワークや LAN などの事例をあげてどのような目的のために利用されているかを紹介する

2. 階層化アーキテクチャ [1]

なぜ階層化の概念が重要なのか．各階層における機能および階層間の関係．TCP/IP，OSI などによって全体の概観を説明する

3. 物理層 [1]

伝送媒体（より対線，同軸ケーブル，光ファイバ，無線，衛星回線），ベースバンド伝送

4. データリンク層 [2]

誤り制御（stop and wait, go back N など），フロー制御（ウィンドウフロー制御），メディアアクセスプロトコル（アロハ方式，LAN における CSMA/CD, トークンパッシング方式など）

事例：イーサネットのしくみと限界，今後の発展（キーワード：より対線による低廉化，10Base-T などのスイッチング技術，光化，高速化，標準化など）

5. ネットワーク層 [5]

パケット交換と回線交換，コネクション型とコネクションレス型通信，輻輳制御，経路制御，マルチメディア通信と品質保証

事例：IP のしくみと限界，今後の発展（キーワード：アドレス構造，インターネットの歴史，標準化，経路制御の方法，なぜコネクションレスか？）

6. ネットワーク管理 [1]

ネットワーク管理の役割と仕組み

事例：SNMP，ARP，DNS（キーワード：ネットワークの大規模化，標準化）

7. トランスポート層 [3]

フロー制御（ウィンドウフロー制御，レート制御），輻輳制御

事例：TCP のしくみと限界，今後の発展（キーワード：ウィンドウフロー制御を用いた輻輳制御のしくみ，マルチメディア通信品質の保証，標準化）

8. アプリケーション層 [2]

下位層プロトコルに基づいたアプリケーションの実現の仕組み，クライアント・サーバモデル，ファイル転送プロトコルと遠隔ターミナル機能，ネットワークファイルシステム，電子メール，WWW

事例：最先端の技術動向，例えば，1999 年なら WWW の動向

先修科目

デジタル通信，コンピュータアーキテクチャ，オペレーティングシステム，信号処理

後修科目

ヒューマン - コンピュータ・インタラクション，ネットワークアーキテクチャ*，分散処理システム*

（*は大学院科目）

関連科目

画像情報処理，情報セキュリティ*

実験及び演習

- TCP/IP を用いたネットワーク環境の構築実習

ワークステーションを与え，ネットワーク構築に必要な管理項目の設定を行なう．例えば，ワークステーションの物理的なネットワーク接続，ルータによるネットワーク間接続，モデム接続，NFS / NIS / メール / ニュースの設定，WWW サーバ構築などが考えられる．オペレーティングシステムの構築実習を行なう場合には，それと共に実施するのも可能．

所要時間：10～15 時間

- ネットワークプログラミング基礎；ソースコードを読む

例えば TCP や http のソースコードを読み，その基本機能を調べる．C 言語の修得が前提．

所要時間：10～15 時間

- ソケットを用いたプロセス間通信に関する計算機演習

ネットワークプログラミング演習によってネットワークの基本概念を体感させる．以下に例題を挙げるが，講義でカバーし切れない場合は演習時間内で補足説明を行なう必要がある．

1) ファイル転送（または計算機間のファイルの複写）を行なうプログラムを TCP/UDP を用いて作成し，コネクション型 / コネクションレス型通信の違いを知る．アクセス制御についても考慮させることが望ましい．

所要時間：10～15 時間

2) 音声パケット通信を実現するプログラムを TCP/UDP を用いて作成し，通信品質の保証がいかに困難かを知る．また，品質を向上させるにはどうすれば良いかを考えさせる．

所要時間：10～15 時間

実施方法

- 演習を行なうのであれば，通年として前期は講義，後期に演習という形が望ましいが，工夫次第で並行に行なうのは可能であろう．なお，演習実施にプログラミング経験は必須である．

教科書/参考書

- コンピュータネットワーク，宮原秀夫，尾家祐二，森北出版，1992（教科書）

コンピュータネットワークの平易な入門書．

- マルチメディア情報ネットワーク — コンピュータネットワークの構成学 —，村田正幸，共立出版，1999 年 6 月刊行予定（教科書）

コンピュータネットワーク入門書．特にマルチメディア通信について丁寧に述べられている．

- Computer Networks; Third Edition, Andrew S. Tanenbaum, Prentice-Hall International, 1996（教科書）

コンピュータネットワークに関する諸技術が幅広く触れられている．ただし，英語．

- Fundamentals of Telecommunication Networks, Tarek N. Saadawi and Nistafa H. Ammar, Wiley Interscience, 1994 (教科書)
基本概念が説明されているだけでなく、最新の技術動向も反映された好書である。
- TCP/IP Illustrated, Volume 1: The Protocols, W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (参考書, 邦訳あり)
- TCP/IP Illustrated, Volume 2: The Implementation, Gray R. Wright and W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (教員用参考書, 邦訳あり)
- TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP, and UNIX Domain Protocols, W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (教員用参考書, 邦訳あり)
以上の3冊のうち、概念を修得するなら第1部のみでよい。また、ソースコードレベルまでの修得を目指すのであれば第2部も必要。最新の技術を反映したものは第3部に示されている。
- Internetworking with TCP/IP Volume I - Principles, Protocols, and Architecture -, Douglas E. Comer, Prentice-Hall International, 1991 (参考書)
- Internetworking with TCP/IP Volume II - Design, Implementation, and Internals -, Douglas E. Comer and David L. Stevens, Prentice-Hall International, 1994 (教員用参考書)
- Internetworking with TCP/IP Volume III - Client-Server Programming and Applications -, Douglas E. Comer and David L. Stevens, Prentice-Hall International, 1994 (教員用参考書)
以上の3冊は邦訳「TCP/IPによるネットワーク構築」(共立出版)がある。第1巻はTCP/IPプロトコル体系の概念をまとめたものであり、第2巻では内部構造を解説している。ネットワークプログラミングを行なうのであれば第3巻も必要であろう。
- インターネット参加の手引き, WIDE Project 編, 共立出版, 1995 (参考書)
- TCP/IP Network Administrations, Craig Hunt, O'Reilly & Associates, 1992 (教員用参考書) (邦訳「UNIXシステム管理者のためのTCP/IPネットワーク管理」トムソン・パブリッシング, 1994)
上記の2冊は、演習でネットワーク構築を行なう際に有用となる。

備考 (コメント)

- 講義項目の順番はプロトコル階層の下位層から始めるという、オーソドックスなアプローチに従っている。しかし、必ずしもその必要はない。特に学生の学習の動機付けのためにも、上位層から始める方がむしろ好ましい。2番目に挙げた教科書はそのような構成をとっている。その場合、項目1, 2を終えた後、最後の項目から進めていけばよい。
- デジタル通信を「ネットワーク入門」科目として扱う場合には、この科目で物理層に関する講義も行なう必要がある。ただし、すべてをカバーできなければ省略してもよい。プロトコルの上位層から始めるのは、そのような技術的な講義の進め方としても意味がある。

U-13 ソフトウェア工学 [1]

目的

ソフトウェア工学の入門として、ソフトウェアを構成するプログラムの各種性質について分析論証できるように、その基礎となる概念や素養を習得させる。また、プログラム設計の手法を具体的に提示して、高品質のプログラムを効率的に開発する手法および支援ツールの必要性と有効性を理解させる。

概要

プログラムそのものの構造および性質、ならびにプログラム開発工程について述べ、高品質のプログラムを効率良く開発するために有用な概念や基礎知識を学習する。プログラム開発の上流工程の重要性を認識させ、特に仕様記述と設計について、いくつかの手法を紹介する。

[修得項目]

仕様記述，信頼性

[知識項目]

段階的詳細化，構造化設計，オブジェクト指向設計

[関連項目]

ソフトウェア開発プロセス，ツールと環境

講義項目

1. プログラムの性質 [1]

プログラムの構造，信頼性，効率，移植性，再利用性，保守容易性

2. プログラム開発工程 [1]

要求定義，仕様記述，設計，実現，保守，運用

3. 要求と仕様 [1]

要求分析，問題の定式化，仕様記述，プロトタイプング

4. 形式的仕様記述 [3]

仕様記述言語，抽象化レベル

5. プログラム設計 [1]

段階的詳細化，アーキテクチャ設計，アルゴリズム設計，データ構造設計，CASE

6. 構造化設計 [3]

データフロー図，実体関連図，データ辞書，CASE ツール

7. オブジェクト指向設計 [3]

オブジェクト，クラス，クラス階層，部品化，再利用，CASE ツール

8. 検証とテスト [2]

正当性の証明，テスト，テストケース，網羅性

先修科目

プログラミング言語論

後修科目

ソフトウェア工学特論

関連科目

データベース，オペレーティングシステム，コンパイラ，数理論理学

実験及び演習

● システム開発演習

授業登録システム，図書貸し出しシステム，ゼミ室予約システムなど身近な問題を題材にして，小規模のシステムの仕様記述と設計，ならびに実現を行なわせる．4～5名程度の小人数で共同開発を行なわせ，仕様記述の重要性ならびに統合テストやシステムテストの難しさを体験させる．グループ毎に異なる設計手法を用いて，それぞれの特徴を報告し合って相互の比較を行なうことにすれば，学生同士での自発的な議論を促進して，興味を持たせることに効果が期待できよう．

実施方法

- 半期で上記の時間配分で教えるならば，個々の項目についてざっと紹介する概論的なものとなるだろう．
- 6, 7 の設計手法を支える基礎として，4, 5, 8 を詳しく教えて，6, 7 については，それぞれ 1 回程度で予告編程度にとどめておくこともできよう．
- 逆に，6, 7 (のいずれか) に重点を置いて，豊富な実例に則して設計手法を詳しく教えても良いが，その場合には，手法が陳腐化しうることを考慮して，できるだけ本質的な概念を習得させるように留意する必要がある．

教科書/参考書

- Niklaus Wirth : Systematic Programming – An Introduction –, B. G. Teubner, 1972. 野下浩平，笈捷彦，武市正人 訳：系統的プログラミング / 入門，近代科学社，1975. (参考書)
系統的な方法でアルゴリズムを構成し定式化する技法としてプログラミングを見なす．古典的な名著である．
- David Gries : The Science of Programming, Springer-Verlag, 1981.
笈捷彦 訳：プログラミングの科学，倍風館，1991. (参考書)
プログラミングを理論や原理に基づく科学の一学問として取り扱った入門書．基礎概念の紹介から始めて具体的なプログラミング方法の提示を行う良書である．教科書として用いても良い．
- Edsger W. Dijkstra and W. H. J. Feijen : A Method of Programming, Addison-Wesley, 1988.
玉井浩 訳：ダイクストラ プログラミングの方法 – 論理学を用いた正しいプログラムの作り方 – ，サイエンス社，1991. (参考書)
上記 2 冊の本に述べられているような方法に関する具体的例題集．

- Carroll Morgan : Programming from Specifications, Second Edition, Prentice Hall, 1994. (参考書)
 仕様から最終的なプログラムへと段階的に詳細化を行う手法を系統的に述べた本である . Gries の本の続編として意欲のある学生に勧めるとよいであろう .
- Michael Jackson : Software Requirements & Specifications - a lexicon of practice, principles and prejudices -, Addison-Wesley, 1995.
 玉井, 酒匂 訳 : ソフトウェア博物誌 - 世界と機械の記述 -, トッパン, 1997. (参考書)
 事典的な構成で , ソフトウェア工学の問題や , それらを解決するための有益な概念および手法を簡潔明解に記述 .
- J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy and W. Lorensen : Object-Oriented Modeling and Design, Prentice-Hall, 1991.
 羽生田栄一 監訳 : オブジェクト指向方法論 OMT : モデル化と設計, トッパン, 1992. (参考書)
 ソフトウェア工学全体というコンテキストの中でプログラム設計を学ぶのに良い .
- 松本正雄, 小山田正史, 松尾谷徹 : ソフトウェア開発・検証技法, 電子情報通信学会, 1997. (参考書)
 ソフトウェア開発, 運用, 保守の全過程における技術と支援環境を取り上げて , その背景や各技法の意義などを実践的な観点から簡潔に解説する .

U-14 数値計算 [1]

目的

数値計算のために考案された各種のアルゴリズムを学ばせると共に、数値計算の結果得られた数値に含まれる誤差の解析と推定法を会得させる。はじめに、数値計算に用いられる浮動小数点数の性質を述べる。浮動小数点数による演算では誤差が生ずるため、数学的には正しい事柄も成り立たないことがあることを説明する。とくに桁落ちによって誤差が急激に増大することを強調する必要がある。桁落ちを防止することによって、はじめて安定なアルゴリズムが得られる。各講義項目で取り上げるアルゴリズムは基本的なものに止める。アルゴリズムを得るための考え方を説明し、誤差と安定性について述べる。誤差を推定するためには、丸め誤差なら計算桁数を変えて計算し、離散化誤差なら離散化パラメータを変えて計算する必要がある。誤差の推定ができてはじめてその数値は意味のある数値となる。この点は繰り返し注意しなければならない。

概要

連立一次方程式、行列の固有値問題、非線形方程式、常微分方程式などを数値的に解くための数値解法や積分を数値的に計算するための方法を述べる。また、その際に生ずる誤差とその推定法について説明する。

[修得項目]

浮動小数点数と内部表現、丸め誤差とその推定法、桁落ちとその防止法、連立一次方程式、ガウスの消去法、ピボット選択と桁落ちの防止、良条件と悪条件、行列の固有値と固有ベクトル、特性方程式、相似変換、ヤコビ法、べき乗法、ハウスホルダー法と桁落ち防止、二分法、逆反復法、非線形方程式、ニュートン法、連立非線形方程式、ヤコビアン行列、多項式補間、ラグランジュ補間、数値積分、ニュートン・コーツ公式(台形則、シンプソン則)、リチャードソンの補外法による離散化誤差の推定、常微分方程式、差分法、オイラー法と後退オイラー法、離散化誤差と安定性

[知識項目]

後退誤差解析、反復解法(ガウス・ザイデル法、SOR法)、加速パラメータ、シュールの定理、ニュートン法の初期値の選択、2次収束、ルンゲの現象、チェビシェフ補間、ロンバーグ積分法、ルンゲ・クッタ法、アダムス法

[関連項目]

コレスキー法、CG法、QR法、ランチョス法、代数方程式とデュラン・ケルナー法、スプライン補間、ガウス型公式、二重指数関数型公式、硬い方程式、ギアの公式

講義項目

1. 数の表現と誤差 [2]

浮動小数点数、IBM形式とIEEE形式、指数部と仮数部、正規化、オーバフロー、アンダーフロー、丸め誤差とその推定法、計算機イプシロン、桁落ちと情報落ち、後退誤差解析

2. 連立一次方程式 [3]

消去法(ガウス法、ガウス・ジョルダン法)、LU分解、ピボット選択、条件数、残差と誤差、反復解法(ガウス・ザイデル法、SOR法)、加速パラメータ

3. 行列の固有値問題 [3]

特性方程式，相似変換，シュールの定理，ヤコビ法，べき乗法，ハウスホルダー法，二分法，スツルムの定理，ゲルシュゴリンの定理，逆反復法

4. 非線形方程式 [1]

二分法，ニュートン法，2次収束，収束判定，連立非線形方程式，ヤコビアン行列

5. 補間 [1]

多項式補間，ラグランジュ補間，ルンゲの現象，チェビシェフ補間

6. 数値積分法 [2]

ニュートンコーツ公式（台形則，シンプソン則），離散化誤差の推定，リチャードソンの補外法，ロンバーグ積分法

7. 常微分方程式 [3]

初期値問題，差分法，オイラー法，後退オイラー法，ルンゲ・クッタ法，アダムス法，安定性，絶対安定領域，離散化誤差の推定

先修科目

プログラミング言語論

後修科目

数理計画法，信号処理

関連科目

なし

実験及び演習

- 講義と並行した実験科目において，講義で習ったアルゴリズムを用いて数値計算を行い誤差と安定性についての理解を深める．

実施方法

- できるだけ数値例を用いて説明すること．
- 簡単な問題を，その場で受講者に解かせると効果的である．

教科書/参考書

- 数値解析，一松信，朝倉書店，1982（参考書）
この講義で扱う項目はすべて含まれている．一部に高度な内容もある．
- 数値解析入門，山本哲朗，サイエンス社，1976（参考書）
この講義で扱う項目はすべて含まれている．とくに非線形方程式に関する記述が詳しい．境界値問題についても述べている．

- 数値解析の基礎，篠原能材，日新出版，1978（参考書）
この講義で扱う項目はすべて含まれている．記述は非常に詳しい．フーリエ展開，境界値問題についても述べている．
- 数値解析講義，F. John，藤田宏・名取亮訳，産業図書，1976（参考書）
原著は1967年発行で少し古いが，定評のある教科書である．誤差の理論的評価に重点を置いている．数値積分はない．その代り放物型，双曲型，楕円型の偏微分方程式を扱っている．
- 数値解析法，森正武，朝倉書店，1984（参考書）
関数計算（補間を含む）と数値積分法が詳しい．非線形方程式と常微分方程式はない．
- 数値計算プログラミング，森正武，岩波書店，1986（参考書）
アルゴリズムの説明だけでなく，FORTRANによるプログラムが載っている．この講義と並行する実験において参考にすると良いだろう．有限要素法，境界要素法，等高線のプログラムなどもある．
- 線形計算，名取亮，朝倉書店，1993（参考書）
連立一次方程式と行列の固有値問題に対するアルゴリズムを数値例を豊富に示しながら丁寧に述べている．自習書として最適であろう．
- 数値計算，伊理正夫，朝倉書店，1981（参考書）
非線形方程式と代数方程式だけを扱っている．数値例が豊富であり，読者は追試計算してみることによって理解を深めることができる．
- 数値解析入門，三井斌友，朝倉書店，1985（参考書）
補間と常微分方程式だけを扱っている．わが国で研究された新しい成果も取り入れられている．
- 数値計算の常識，伊理正夫・藤野和建，共立出版，1985（参考書）
通常の教科書とは異なり，数値計算を行う上で心得ておくべき重要なポイントを取り上げて具体的な数値例を示しながら述べている．この本に書いてあることが「常識」となれば一人前というわけである．

U-15 集積回路工学 [1]

目的

デジタル集積回路の製造法，設計法，設計ツール，設計フローに関して常識的な知識を持たせる．選択科目でよい．学科の特徴に応じて，簡単な基本知識を身に付けさせる立場から，実際に設計ができる人を育てるための実務的な科目と位置付ける立場まで柔軟に考えて欲しい．必ずしもすべての項目をカバーしなくてもよい．

概要

現在の CS を支えている要素技術である集積回路に関する基本的な知識を講義する．特に，デジタル集積回路の製造法，設計法，設計ツール，設計フローに関する基礎的な知識を理解し，CS の道具として集積回路技術を利用できる基本的な技術を修得する．

[修得項目]

メモリ集積回路の構造，論理集積回路の構造，集積回路設計の流れ，論理合成，シミュレーション，テスト生成，レイアウト生成

[知識項目]

集積回路の技術の変遷，CMOS の製造工程，集積回路の経済，速度/面積/電力のトレードオフ，テスト容易化設計

[関連項目]

CMOS 以外の技術，アナログ回路設計，マイクロプロセッサ設計

講義項目

1. 集積回路の歴史 [2]

集積回路の技術の変遷，回路の実現性を規定する要素（速度，面積，電力）

2. メモリと論理 [2]

メモリ集積回路の構造，論理集積回路の構造，CMOS 以外の技術

3. 集積回路の製造過程 [1]

CMOS の製造工程，製造工程で必要となるデータ

4. 基本回路設計 [2]

基本回路設計のツールと手法，回路シミュレータ，回路抽出，レイアウトエディタ，基本ゲートの設計

5. レイアウト設計 [2]

セルベース設計のレイアウト手法，配置/配線アルゴリズム，配線による遅延と電力消費

6. 論理設計と論理合成 [2]

論理合成のアルゴリズム，ハードウェア記述言語，ゲートアレイとセルベース設計の違い，FPGA の構造と原理

7. アーキテクチャ設計とフロアプラン [2]

マクロセル，階層設計，フロアプランの重要性，設計上位レベルでの性能見積り

8. テストと検証 [2]

テストの手法，テスト生成，テスト容易化設計，シミュレーションによる設計検証，形式的設計検証

先修科目

論理回路，コンピュータアーキテクチャ，論理設計（電子回路）

後修科目

なし

関連科目

信号処理，画像情報処理，音声情報処理

実験及び演習

- 集積回路の製作実験：実際に設計ができる人を育てるための実務的な科目と位置づける場合は，実験や実習と一体化することが望ましい．簡単な集積回路（簡単なマイクロプロセッサなど）を試作させる．東京大学の大规模集積システム設計教育研究センター（VDEC）を利用して，最新の設計ツールを利用する．時間的，予算的な余裕があれば，VDEC を利用した試作を行わせることは極めて大きな効果が期待できる．

実施方法

- 選択科目でよい．実際に設計ができる人を育てるための実務的な科目と位置づける場合は，実習と一体化することが望ましい．必ずしもすべての項目をカバーしなくてもよい．場合によっては，産業界の経験者による数時間の特別講義の形で主な事項を講義することも可能である．電気電子工学系の講義がある場合は，そちらとの関連についても配慮が必要である．

教科書/参考書

- Principles of CMOS VLSI Design – A systems Perspective –, Second Edition, Neil H.E. Weste and Kamran Eshraghian, Addison Wesley, 1993.（教科書）
情報系の学生が知っておくべき集積回路に関する知識が非常にわかりやすく述べられている名著．日本語訳：CMOS VLSI 設計の原理 - システムの視点から - ，富沢，松山監訳，丸善，1988.（ただし，第1版の訳である）．
- CMOS 超 LSI の設計，菅野卓雄監修，培風館，1989（参考書）
- VLSI System Design - When and How to Design Very-Large-Scale Integrated Circuits -, Saburo Muroga, John Wiley & Sons, 1982.（参考書）

U-16 信号処理 [1]

目的

実世界で観測された信号から，いかに有意な情報を抽出するかについて講述する．単に数学的に直交変換を説明するのではなく，時間領域で観測される信号を周波数領域などへ直交変換しても，あくまで信号としての本質が保存されていることを理解させる．そして，実世界の信号や雑音の多くは周波数領域で記述した方がより本質的であることを，実例を以て体得させる．また，信号のデジタル化は，標本化限界，量子化雑音の存在によって困難であることを理解させる．DSP，AD変換器，DA変換器などの信号処理ハードウェアについて語るのではなく，数学的背景と信号の概念修得を中心とする．

概要

デジタル信号処理の基礎知識を修得する．連続信号を離散化，量子化することによる情報の劣化を学び，それとともに離散信号の数学的な表現法から時間領域の波形信号を周波数領域へ変換し，各種周波数領域での分析・処理手法について学ぶ．

[修得項目]

フーリエ級数展開，フーリエ変換，フーリエ変換の性質，離散フーリエ変換，離散フーリエ変換の性質，高速フーリエ変換，信号の時間領域表現と周波数領域表現，パワースペクトル，Z変換，Z変換の性質，伝達関数，インパルス応答，信号の標本化と量子化，標本化定理，折り返し雑音，量子化雑音，周波数解析，窓関数

[知識項目]

フィルタ，畳み込み積分，sinc関数，Gibbs現象，パーシバルの定理，標本化限界，FFTアルゴリズム，自己相関関数，インパルス応答，矩形窓，ハニング窓

[関連項目]

アナログ対デジタル，モデム回路，システム応答，デルタ関数直交変換，アダマール変換，ウォルシュ変換，コサイン変換，デジタル信号プロセッサ(DSP)，AD変換器，DA変換器，情報源圧縮，音響信号処理，画像信号処理，S/N

講義項目

1. 数学的準備 [3]

フーリエ級数展開，Gibbs現象，フーリエ変換，逆フーリエ変換，sinc関数

2. 数学的準備 [1]

信号の時間領域表現と周波数領域表現，パワースペクトル，パーシバルの定理，自己相関関数

3. 信号の離散化 [2]

信号の標本化と量子化，標本化定理，標本化限界，折り返し雑音，量子化雑音

4. 離散フーリエ変換 [1]

離散フーリエ変換，離散フーリエ変換の性質

5. 離散フーリエ変換の計算法 [2]

6. 離散フーリエ変換による信号解析 [2]

周波数解析, 短時間周波数解析, 窓関数, 矩形窓, ハニング窓

7. Z 変換 [2]

Z 変換, 逆 Z 変換, Z 変換の性質

8. システム応答 [2]

伝達関数, システム応答, インパルス応答, フィルタ, 畳み込み積分

先修科目

解析学, 複素関数論, 確率論と情報理論, 数値計算

後修科目

画像情報処理, 音声情報処理

関連科目

パターン認識, デジタル通信, 論理設計, 集積回路工学

実験及び演習

離散フーリエ変換以降の講義項目については, 信号処理ソフトウェアパッケージを用いた実験及び演習を連動させるとよい。特に Gibbs 現象, 折り返しノイズや窓効果を実際に図化させることで体得させる。

- Gibbs 現象: 正弦波の組み合わせから任意の波形信号が合成できることを実証する。同時に Gibbs 現象を理解させる。
- 信号のサンプリング: 与えた連続信号をサンプリングさせ, そのサンプリングレートを変化させることで, 折り返しノイズを発生させ, サンプリングの限界を修得させる。
- 窓: 高速フーリエ変換のソフトウェアパッケージを用いて脳波などの実世界の信号を短時間周波数解析をさせる。その際, 各種の窓によって周波数分解の精度が異なることを修得させる。

実施方法

- 講義時間内に実験や演習を行うと, 時間が足りなくなる恐れがある。その場合には, 離散フーリエ変換の計算法の項目を省略することも考えられる。

教科書/参考書

- デジタル信号処理の基礎, 樋口龍雄 著, 昭晃堂, 1986 (教科書)
- デジタル信号処理の基礎, 前田渡 著, オーム社, 1980 (教科書)
- デジタル信号処理の基礎, 辻井重男 著, 電子情報通信学会, 1988 (教科書)
- デジタル信号処理ハンドブック, 電子情報通信学会 編, 電子情報通信学会, 1993 (教員用参考書)

- 高速フーリエ変換とその応用, 佐川雅彦, 貴家仁志 共著, 昭晃堂, 1992 (参考書)
- ビギナーズデジタル信号処理, 小畑秀文, 幹康 共著, 東京電機大学出版局, 1989 (参考書)
初心者向けの参考書.
- ビギナーズデジタルフーリエ変換, 小畑秀文, 幹康 共著, 東京電機大学出版局, 1989 (参考書)
初心者向けの参考書.
- CAI デジタル信号処理, 小畑秀文, 幹康 共著, コロナ社, 1991 (参考書)
初心者向けの参考書.
- Digital Signal Processing, W. Mayeda 著, Prentice-Hall, 1993 (教員用参考書)

U-17 画像情報処理 [1]

目的

2次元信号(パターン)のデジタル処理を扱うデジタル画像処理の基本的な概念と方法論について講述する。特に、画像のデジタル化と信号レベル/画素レベルの処理を理解させる。画像処理手法については、様々なアルゴリズムを網羅的に説明するのではなく、処理目的と数学的な裏付けを示し、画像処理アルゴリズムの背景と方法論の修得に重点をおく。また、具体的な応用との関係を理解させる。

概要

デジタル画像処理の基礎知識を修得する。まず画像の標本化と量子化及び符号化を学ぶことによってデジタル画像の性質を理解し、続いて、画像の変換処理と人間の視覚認識機能をコンピュータにもたせようとする画像認識の基礎である画像のセグメンテーションと特徴抽出について学ぶ。

[修得項目]

画像の標本化, 量子化, デジタル画像の性質, 画像の種類, 階調変換, ヒストグラム平坦化, 幾何学的変換, 画像復元, 雑音除去, メディアンフィルタ, 平滑化, ガウシアンフィルタ, 画像強調, ガウシアンラプラシアンフィルタ, DOG フィルタ, エッジ検出, 零交差, Hough 変換, 領域分割, 2 値化, 判別分析法, 幾何学的特徴, 2 値画像処理, 細線化, テクスチャ特徴

[知識項目]

色分解, 表色系, 色変換, エッジ保存平滑化, 位置合わせ, 色特徴, フーリエ解析, フーリエ記述子, 動き特徴 (オプティカルフロー), テンプレートマッチング

[関連項目]

静止画像の符号化, 動画画像の符号化, 知的符号化, 符号化方式の標準化, JPEG, MPEG, 画像のフラクタル特徴

講義項目

1. 画像のデジタル化 [2]

画像の標本化, 量子化, デジタル画像の性質, 画像の種類, 色分解, 表色系

2. 画像の符号化 [2]

静止画像の符号化, 動画画像の符号化, 知的符号化, 符号化方式の標準化, JPEG, MPEG

3. 画像の変換処理 [4]

階調変換, ヒストグラム平坦化, 色変換, 幾何学的変換, 画像復元, 雑音除去, メディアンフィルタ, 平滑化, ガウシアンフィルタ, エッジ保存平滑化, 画像強調, 位置合わせ

4. 画像のセグメンテーション [3]

エッジ検出, ガウシアンラプラシアンフィルタ, DOG フィルタ, 零交差, Hough 変換, 領域分割, 2 値化, 判別分析法

5. 特徴抽出 [4]

幾何学的特徴, 2 値画像処理, テクスチャ特徴, 画像のフラクタル特徴, 色特徴, フーリエ解析, フーリエ記述子, 動き特徴 (オプティカルフロー), テンプレートマッチング

先修科目

確率論と情報理論, 信号処理, パターン認識

後修科目

画像情報処理特論

関連科目

情報代数と符号理論, 数理計画法, 数値計算, デジタル通信, コンピュータグラフィックス, ヒューマン-コンピュータ・インタラクション, 人工知能, 音声情報処理, 集積回路工学, 情報ネットワーク

実験及び演習

- 画像処理演習: 3~5の講義項目については, 画像処理ソフトウェアパッケージを用いた演習を連動させるとよい。

実施方法

- 講義時間内に実験や演習を行うと, 時間が足りなくなる恐れがある。その場合には, 項目2を省略し, 講義内容を画像のデジタル化とデジタル画像処理の基本手法に限定することも考えられる。
- 講義項目と関連づけて具体的な応用事例を紹介するとよい。

教科書/参考書

- Digital Image Processing, A. Rosenfeld and A.C. Kak, Academic Press, 1976 (教科書, 教員用参考書)
デジタル画像処理, 長尾真 監訳, 近代科学社, 1978
デジタル画像処理の概念とアルゴリズムが体系的にまとめられている。
- Digital Image Processing (2nd Edition), A. Rosenfeld and A.C. Kak, Academic Press, 1982 (教員用参考書)
上記の改定版。
- 画像認識論, 長尾真 著, コロナ社, 1982 (教科書)
画像処理・認識について簡潔に書かれている。
- コンピュータ画像処理入門, 田村秀行 監修, 総研出版, 1985 (参考書)
デジタル画像処理の基本概念, 手法, 応用システム事例が簡潔にまとめられている。続編として応用実践編も出ている。
- 画像処理アルゴリズムの最新動向, 高木幹雄, 鳥脇純一郎, 田村秀行 編, 新技術コミュニケーションズ, 1986 (教員用参考書)
デジタル画像処理全般のアルゴリズムに関する分野別サーベイ集であり, 3次元再構成, 動画像処理を含むとともに, いくつかの応用分野ごとのサーベイも行っている。
- 画像認識の基礎 [I], [II], 森俊二, 坂倉梅子 共著, オーム社, 1986, 1990 (参考書)
特徴抽出のための画像処理手法が詳しく述べられている。

- 画像処理ハンドブック, 尾上守夫 編, 昭晃堂, 1987 (教員用参考書)
デジタル画像処理全般の手法・アルゴリズムが網羅的に示されている.
- 画像理解のためのデジタル画像処理 [I], [II], 烏脇純一郎 著, 昭晃堂, 1988 (教科書)
デジタル画像処理の基本手法を厳密な定義に基づいて紹介しており, 特に, 2 値画像処理に詳しい.
- Digital Image Processing (2nd Edition), W.K. Pratt, Wiley, 1991 (教員用参考書)
デジタル画像処理アルゴリズムについてまとめており, 特にフィルタリングについては詳しく書かれている.

備考 (コメント)

- 上に示した教科書/参考書は比較的古いものが多いので, 補助教材等を用いて新しい応用事例を紹介することによって最近の動向を教えるとよい.

U-18 パターン認識 [1]

目的

一般的なパターン認識系の構成，及びベイズ決定論 (情報理論) に基づく識別機構を講述する．識別関数の各クラスについて，複雑度 (パラメータ数) と，識別能力 (closed サンプルの性能) と頑健性 (open サンプルの性能) の関係を論じる．この観点から，統計的モデルに基づくパラメトリック識別系と，識別学習に基づくノンパラメトリック識別系の構成法を述べ，両者の比較も行う．

概要

一般的なパターン認識系の構成，及びベイズ決定論 (情報理論) に基づく識別機構について学ぶ．一般的な識別関数について各クラスの特徴を比較する．その具体的な構成法として，統計的モデルに基づくパラメトリック識別系と識別学習に基づくノンパラメトリック識別系について修得する．

[修得項目]

パターン認識系の構成，ベイズ決定論，A-D 変換，統計的特徴抽出，線形識別，2 次識別，区分線形識別，機械容量，正規分布に基づく統計的識別モデル，ニューラルネットワーク (NN)，動的時間正規化 (DTW)

[知識項目]

サンプリング定理，量子化歪，判別分析，K-L 変換，ベクトル量子化 (VQ)，最尤推定，対角共分散正規分布，等共分散正規分布，混合正規分布，最小 2 乗誤差 (MSE) 学習，バックプロパゲーション，DP マッチング，隠れマルコフモデル (HMM)，トレリス計算

[関連項目]

音声符号化，クラスタリング，K-means，最小識別誤り (MCE) 学習，前向きアルゴリズム，Viterbi アルゴリズム，教師付き/教師なし学習

講義項目

1. パターン認識系 [2]

観測，前処理，特徴抽出，識別判定，ベイズ決定論

2. A-D 変換と符号化 [1]

サンプリング定理，量子化歪，AD-PCM， μ -LAW

3. 統計的特徴抽出 [2]

判別分析，K-L 変換，ベクトル量子化 (VQ)

4. 識別関数のクラス [2]

線形識別，2 次識別，区分線形識別，K-means，機械容量

5. パラメトリック識別系 (統計的識別モデル) [3]

正規分布に基づく最尤推定，対角共分散分布，等共分散分布，混合正規分布

6. ノンパラメトリック識別系 (識別学習) [3]

最小 2 乗誤差 (MSE) 学習, ニューラルネットワーク (NN), バックプロパゲーション,
最小識別誤り (MCE) 学習

7. 時系列パターンの識別 [2]

DP マッチング, 隠れマルコフモデル (HMM), トレリス計算, 前向きアルゴリズム,
Viterbi アルゴリズム

先修科目

情報代数と符号理論, 確率論と情報理論, 数理計画法

後修科目

画像情報処理, 画像情報処理特論, 音声情報処理

関連科目

ディジタル通信, 信号処理, 人工知能, 自然言語処理, ヒューマン-コンピュータ・インタラクション

実験及び演習

- 音声 (母音程度) や文字 (数字程度) を対象にして, 簡単なパターン認識系を構成する. サンプルングや量子化の粒度による違いを体験する. 種々のクラスの識別関数を構成し, closed サンプルと open サンプルに対する識別率を評価することにより, 識別性能や頑健性などを比較する. また, 学習サンプル数とパラメータ数の関係を考察する. なお, 実験及び演習は講義とは別 (の時間枠) に行うことが望ましい.

実施方法

- 後半の 3 つの講義項目については, 講師の専門に応じて, 項目や比重が変更可能である.

教科書/参考書

- Learning Machines, N.J.Nilsson 著, Morgan Kaufmann, 1990 (教科書)
この科目の目標に即した教科書である.
- パターン認識, 森健一 著, 電子情報通信学会・コロナ社, 1988 (参考書)
- パターン認識と図形処理 (岩波情報講座情報科学 21), 長尾真 編, 岩波書店, 1983 (参考書)
- 認識工学 -パターン認識とその応用-, 鳥脇純一郎 著, コロナ社, 1993 (参考書)
- 画像認識論, 長尾真 著, コロナ社, 1983 (参考書)
画像の認識を中心としているが, パターン認識の多くの基本的概念が丁寧に述べられている.
- パーセプトロン -パターン認識への道-, ミンスキー, パパート 著, 東京大学出版会, 1971 (参考書)
- わかりやすいパターン認識, 石井健一郎, 上田修功, 前田英作, 村瀬洋 共著, オーム社, 1998 (参考書)
統計的パターン認識の代表的な理論を紹介し, パターン認識と学習の基本的な考え方を解説している.

備考 (コメント)

- 音声や文字，画像などの個々のアプリケーションにおける実現法は「画像情報処理」や「音声情報処理」などで説明されるので，本講義では，できるだけ一般的・普遍的な概念を述べるようにする．

U-19 ヒューマン - コンピュータ・インタラクション [2]

目的

ここでは、コンピュータシステムやコンピュータソフトウェアの利用者インタフェースの設計に必要な知識の習得を目的とする。CS 学科を卒業した学生が、コンピュータシステムのハード/ソフトウェアの設計に携わる場合に必要な知識について習得することを目標とする。コンピュータの利用者インタフェースに関する幅広い議論を中心とし、特定のインタフェースの設計技法（例えば GUI プログラミング）のみの習得ではなく、先進的・実験的なインタフェースの事例などについても積極的に取り上げることが望ましい。

概要

コンピュータシステムやコンピュータソフトウェアの利用者インタフェースの設計に必要な利用者インタフェースの重要性、良いインタフェースの定義、インタフェースの評価手法、設計方法、利用者インタフェースハードウェア、デザインサイクルなどについて講義する。

[修得項目]

ヒューマン - コンピュータ・インタラクションの定義、コンピュータコストとヒューマンコストのトレードオフ、コンピュータ技術の歴史、利用者中心デザイン、ソフトウェア工学との関係、インタフェース評価、GUI、WIMP インタフェース

[知識項目]

コンピュータ利用者層の変遷、コンピュータ機器の変遷、システムへの要求とインタフェースへの要求との関係、視覚、聴覚、刺激と知覚、記憶、学習、アフォーダンス、人間情報処理モデル、GOMS、KL-モデル、文字表示装置、図形表示装置、3次元表示装置、文字入力装置、日本語入力方式、ペン入力方式、ポインティングデバイス、多次元ポインティングデバイス、音声入出力、デザインサイクル、デザインガイド、プロトタイピング、繰り返しデザイン、予測評価手法、実験的評価手法、ユーザビリティ評価、質問表、エラー分析、GUI ツールキット、X-Window プログラミング、ビジュアルプログラミング、コンピュータグラフィックス、Information Visualization(情報可視化)、Information Retrieval、マルチメディアインタフェース、マルチモーダルインタフェース、仮想現実、CSCW、グループウェア、バリアフリーインタフェース

[関連項目]

コンピュータアーキテクチャ、オペレーティングシステム、ソフトウェア工学、プログラミング言語、画像処理、信号処理、音声処理、マルチメディア処理、言語処理

講義項目

1. ヒューマン - コンピュータ・インタラクションとは？ [2]

ヒューマン - コンピュータ・インタラクションの定義、コンピュータコストとヒューマンコストのトレードオフ、コンピュータ技術の歴史（コンピュータ利用者層の変遷、コンピュータ機器の変遷）、システムへの要求とインタフェースへの要求との関係、利用者中心デザイン、アフォーダンス、認知科学との関連、ソフトウェア工学との関連

2. 知覚/認知の基礎知識 [3]

感覚器、受容器、神経細胞、視覚、聴覚、刺激と知覚、注意と記憶、学習、人間情報処理モデル

3. 入出力ハードウェア [2]

文字表示装置，図形表示装置，3次元表示装置，文字入力装置（キーボードとキー配列），日本語入力方式，ペン入力インタフェース，ポインティングデバイス，音声入出力

4. インタフェース設計手法 [2]

デザインガイド，要求分析，タスク分析，要求定義，プロトタイピング，繰り返しデザイン

5. インタフェース評価手法 [2]

予測評価手法（GOMS, KL モデル, NGOMSL など），実験的評価手法（プロトコル解析，ログ分析，パフォーマンス評価など），ユーザビリティ評価（インスペクション法など），質問表，エラー分析，比較評価，ユーザビリティエンジニアリング

6. GUI プログラミング [2]

GUI ツールキット，X-Window プログラミング，ビジュアルプログラミング，コンピュータグラフィックス，

7. 先進的インタフェース [2]

マルチメディアインタフェース，マルチモーダルインタフェース，仮想現実インタフェース，CSCW，グループウェア，情報可視化技術

8. バリアフリーインタフェース [1]

情報処理機器アクセスビリティガイドライン，視覚/聴覚障害者向けインタフェース，

先修科目

コンピュータサイエンス序説

後修科目

なし

関連科目

デ - タ構造とアルゴリズム，プログラミング言語論，コンピュータアーキテクチャ，ソフトウェア工学，オペレーティングシステム，コンパイラ，論理回路，論理設計，データベース，情報ネットワーク，人工知能，画像情報処理，コンピュータグラフィックス，パターン認識，自然言語処理，集積回路工学，ソフトウェア工学特論

実験及び演習

- GOMS/KL モデルによるパフォーマンス予測評価：身近にあるソフトウェアのインタフェース（エディタなど）を GOMS，NGOMSL，KLM などでのパフォーマンス予測をおこなう。
- GUI プログラミング：ある機能を実現するユーザインタフェースを設計し，それを実装/評価する。演習として設計された何種類かのユーザインタフェースの特徴を議論する。

- プロトタイプデザイン：具体的な課題を設定し，ユーザインタフェースを設計し，初期段階での評価を紙芝居方式，あるいはハイパーテキストをもちいて実施する．評価結果を設計にフィードバックし設計変更を繰り返す実験を実施する．

実施方法

- 科目名は，Human-Computer Interaction(HCI) の適切な日本語訳が定まっていないので，そのままカタカナ表記とした．講義は，関連科目を先に習得したあとで実施する場合と，利用者から見たコンピュータについて早い時点で理解させるために他の関連講義に先だてて実施する場合とが考えられる．それぞれの場合で，講義内容は取捨選択してよいが，1,2 および 7, 8 はどちらの場合でも同じ内容で実施することが可能である．3 については場合によっては省略可能であるが，ペン入力などは GUI プログラミングの項目でも触れてもよい．

教科書/参考書

この分野での教科書的な本は日本語では適当なものが見当たらないので以下を参考書としてあげる．

- Human-Computer Interaction, Jenny Preece et al., Addison-Wesley, 1994. (参考書/教員用参考書)
極めて綿密に設計され，精力的に書き下されている．視野を広げる効用が大きい．
- Curricula for Human-Computer Interaction, ACM SIGCHI, ACM, 1992. (教員用参考書)
Preece らの本の構成の基本となった ACM SIGCHI の カリキュラムグループの報告書である．

備考 (コメント)

- コンピュータが身近な道具として利用されることにより，その利用者インタフェースの重要性は増してきている．従来，利用者インタフェースの評価は CS 学科出身者でなく，実験心理学，心理物理などの基礎知識を習得したものが実施すべきとの考えが支配的であったが，コンピュータシステム/コンピュータソフトウェアの開発にかかわるすべてのエンジニアが，HCI に対する共通の認識を持つことがなにより重要である．このような共通の認識なくしては，インタフェースのデザインサイクルを構築することは困難である．このため，HCI に関する教育は大学院でなく学部で実施されることを前提として作成した．

また，日本では，利用者インタフェース設計の現場で CS 出身のエンジニアがそのデザインと実装の多くの部分を担っているため，エンジニア自身が HCI に対しての理解だけでなく評価手法の実際などを習得しておくことは有用であると考えられる．したがって，本講義では実習を行なうことを勧める．

U-20 コンピュータグラフィックス [2]

目的

画像の出力処理という立場から画像の提示方法，3次元空間の扱い方，3次元を2次元に変換する幾何学変換の方法，などについて学習させることにより，コンピュータで画像を生成する原理，方法を理解させる。

概要

画像を使ってコンピュータとコミュニケーションを行うとき，コンピュータ側からの出力のためには，画像を生成しなければならない。本科目では，画像を生成する基礎理論，画像を表示する方式について学習する。

[修得項目]

画像生成モデル，光源モデル，反射モデル，物体の記述方法，エリアジング，クリッピング，ビューイング，隠線（面）消去

[知識項目]

画像出力装置，グラフィックスライブラリ，3次元から2次元への変換方法，ステレオの原理，人間の視覚システム，カメラモデル

[関連項目]

コンピュータビジョン，CADシステム，グラフィックス専用ハードウェア，画像処理ライブラリ，画像処理方式，パターン認識，音声情報処理，音響処理，自然言語処理，マルチメディア処理，認知科学，視覚心理学

講義項目

1. コンピュータグラフィックスの基礎 [4]

画像生成モデル，光源モデル，カメラモデル，形の記述法，色表現，陰影表現，テクスチャ表現，動きの表現法

2. 画像出力の原理 [2]

画像出力装置，グラフィックスソフト，ラスタ走査，ビットマップ

3. 2次元空間での図形生成 [4]

デジタル幾何学，線の描画法，画像の扱い方，疑似濃淡表示，色表示法，形の表示法

4. 2次元空間からの画像生成 [2]

エリアジング，ビューイング，クリッピング，隠線消去

5. 3次元空間での立体生成 [4]

立体の表現法，ソリッドモデル，パッチモデル，CSG，幾何学変換，透視変換

6. 3次元空間からの画像生成 [4]

レンダリング，テクスチャマッピング，バンプマッピング，表面色の表示，照明の扱い，視点変換，隠面消去，投影法

先修科目

離散数学, データ構造とアルゴリズム, データベース, 解析力学, 計算幾何学

後修科目

ヒューマン - コンピュータ・インタラクション, 認知科学, 認知心理学

関連科目

画像情報処理, 音声情報処理, 自然言語処理, 芸術関連科目

実験及び演習

- 画像生成演習：上記関連科目の実験及び演習と統合させると良い。

実施方法

- すべての項目を行なうと2半期必要であるが, もし, 半期で本科目を実施するならば, 1の全項目と3, 4, 5の一部を最低限学習するのが望ましい。

教科書/参考書

- 河合さとる：基礎グラフィックス, 昭晃堂, 1985。(教科書)
グラフィックス全体に関して原理と応用が述べられている。
- 中前栄八郎, 西田友是：3次元コンピュータグラフィックス, 昭晃堂, 1986。(教科書)
3次元物体の表示方法を中心に解説してある。
- Computer Graphics principles and practice, Foley et al., Addison Wesley, 1992。(参考書)
原理と応用が詳しく述べられている。
- T. Pavlidis : Algorithms for Graphics and Image Processing, Computer Science Press, 1982。(参考書)
画像処理とグラフィックスの具体的なアルゴリズムが述べられている。

備考 (コメント)

- できれば独立した科目としたいが, やむを得ない場合は「画像情報処理」と一緒に科目設計してもよい。

U-21 自然言語処理 [1]

目的

人間の言語，特に書き言葉の理解に対するコンピュータ処理の方法および問題について理解させることを目的とする．英語あるいは日本語を題材として，形態素解析，構文解析，意味解析，文脈解析など個々のコンポーネントにおける基本的な表現法やアルゴリズムおよび理論的な基礎について講述する．文法理論とそれらのコンピュータ上での実現および基本となるアルゴリズムについての講義を中心とするが，その他のトピックを適宜含めることも考えるべきである．その方法としては，(1) 文法理論や意味論などの理論的な側面を重視する，(2) 電子化辞書や電子化テキスト (コーパス) を利用した最近の研究，特に統計的手法を詳述する，(3) 文脈処理，文生成などを含め，自然言語処理にまつわる基本的な技術の概要を網羅的に解説する，(4) 機械翻訳，情報検索などの実用的な応用を中心に種々の技術の紹介を行なう，などが考えられる．

概要

コンピュータによる自然言語処理の基本的な技術と応用について講述する．形態素解析，構文解析の代表的なアルゴリズム，理論的な文法体系，語に関する知識，言語処理における曖昧性などの基本的な問題について説明する．また，機械翻訳などの応用技術についても紹介する．

[修得項目]

形態素解析，文脈自由文法，代表的な構文解析アルゴリズム，曖昧性の問題，文法理論

[知識項目]

個々の構文解析アルゴリズム，Chomsky 文法，X バー理論，語彙機能文法，主辞駆動句構造文法，素性構造と単一化，統計的品詞タグ付け，確率文法，機械翻訳，情報検索

[関連項目]

講義項目

1. 自然言語処理概要 [2]

文字処理，言語のエントロピー，かな漢字変換，形態素解析，構文解析，意味解析，語用論，自然言語処理における辞書と文法，曖昧性の問題

2. 形態素および構文解析アルゴリズム [5]

形態素解析，文脈自由文法，構文解析アルゴリズム (トップダウン法，shift-reduce 法，left-corner 法，CKY 法，Earley 法，Chart 法，一般化 LR 法)

3. 文法理論 [2]

生成文法 (原理とパラメータに基づく文法理論)，語彙機能文法，主辞駆動句構造文法，素性構造と単一化

4. 辞書と語彙知識 [2]

シソーラス，格フレーム，電子化辞書，コーパス

5. 統計的自然言語処理 [2]

共起関係，統計的品詞タグ付け，確率文法とパラメータ学習，意味クラスタリング

6. 自然言語処理応用 [2]

データベースアクセス，機械翻訳，情報検索，情報抽出

7. (意味論 [1])

モンタギュ文法，状況意味論，種々の知識表現法，語彙意味論

8. (語用論 [2])

文脈処理概要，照応関係，スクリプト，プラン，ゴール，Grice の会話の公準，関連性理論

9. (文生成 [1])

テンプレート，システミック文法，意味主辞駆動生成

先修科目

形式言語とオートマトン，データ構造とアルゴリズム，数理論理学

後修科目

なし

関連科目

人工知能，音声情報処理

実験及び演習

- 簡単な構文解析あるいは機械翻訳などの応用プログラムの試作

実施方法

- 任意の言語を用いて上記プログラムを作成する．あるいは，簡単な文法を Prolog に組み込みの DCG(Definite Clause Grammar) で記述しておき，それを学生に拡張させる (Prolog プログラミングが先修条件)

教科書/参考書

- 自然言語理解，田中穂積，辻井潤一，オーム社 知識工学講座 8，1988 (教科書)
自然言語処理に関連する主な理論から処理について簡潔にまとめられている．
- 自然言語処理，長尾真，岩波書店，1996 (教科書)
主に処理の側面から詳細にまとめられた入門書
- 自然言語解析の基礎，田中穂積，産業図書，1989 (参考書)
形態素解析および構文解析アルゴリズムについて詳細にまとめられている
- 自然言語，郡司隆男，日本評論社，1994 (参考書)
自然言語の文法理論について簡潔にまとめられた教科書

- Natural Language Understanding (Second Edition), James Allen, The Benjamin/Cummings Publishing, 1995 (教員用参考書)
自然言語処理のほとんどの側面について詳細にまとめられた教科書．半期の授業では分量が多過ぎるので，適当な章を選択して教えるための参考書となる．
- 言語の数理，岩波講座「言語の科学」第8巻，長尾真 他 編集，岩波書店，1998
形式言語の基礎的な内容から自然言語のための文法記述システムや統語解析，文生成について簡潔にまとめられている．

大学院科目 (学部科目の特論) A

論理回路特論 A-1

コンピュータアーキテクチャ特論 A-2

オペレーティングシステム特論 A-3

データベース特論 A-4

画像情報処理特論 A-5

コンピュータグラフィックス特論 A-6

信号処理特論 A-7

ソフトウェア工学特論 A-8

A-1 論理回路特論 [1]

目的

論理回路に関する専門的な知識を整理するとともに，集積回路設計への応用の立場および並列計算の数学モデルとしての立場からの議論を展開する．内容は，担当者の選択に大きく依存する．ここで示すものは 1 例である．

概要

論理関数に関して学部より先端的な部分を講義し，計算機科学の基礎的な道具としての論理回路に関する理論を整理する．実用的な集積回路設計の立場からは，算術演算回路に関する効率の良い種々の回路/アルゴリズムの紹介や，論理合成および論理検証に利用できる諸知識を整理する．また，並列計算モデルとしての論理回路の立場からも，論理関数の複雑さの理論などを講義する．

[修得項目]

論理関数の表わし方，完全性，不完全指定順序回路の設計法，非同期論理回路設計法，加算回路の設計，乗算回路の設計，論理関数の複雑さ，NP 完全問題，P 完全問題

[知識項目]

グラフ表現 (2 分決定図など) ，幾何学的表現，線形関数，しきい関数，対称関数，ユネイト関数，正関数，双対関数，積和形，スピード非依存回路

[関連項目]

集積回路設計，並列アルゴリズム設計，ハードウェアアルゴリズム

講義項目

1. 論理関数とその表現 [2]

論理関数の表わし方，論理式，グラフ表現 (2 分決定図など) ，幾何学的表現

2. 論理関数の諸性質 [2]

完全性，線形関数，しきい関数，対称関数，ユネイト関数，正関数，双対関数，積和形

3. 順序回路設計法 [2]

不完全指定順序回路の設計法，シフトレジスタ実現

4. 非同期論理回路設計法 [2]

競合とハザード，状態割り当て法，スピード非依存回路

5. 基本回路の設計 [2]

加算回路の設計，乗算回路の設計

6. 論理関数の複雑さ [2]

並列計算モデルとしての論理回路，素子数と段数，計算量と計算時間，式の長さ

7. NP 完全問題と P 完全問題 [2]

NP 完全問題，充足可能性判定問題，P 完全問題，回路値問題

教科書/参考書

- 論理設計スイッチング回路理論，笹尾勤，近代科学社，1995. (教科書)
論理関数の性質について詳しい．理論的に良く整理された教科書である．
- 論理回路，高木直史，New Text 電子情報系シリーズ，昭晃堂，1997 (教科書)
古典的な論理回路に関する事項をコンパクトにまとめて平易に解説している．
- Introduction to Digital Logic Design, J.P. Hayes, Addison Wesley, 1993. (教科書)
- Digital Design -Principles and Practice - 2nd. Ed., J. Wakerly, Prentice Hall, 1994. (教科書)
- Principles of Digital Design, Daniel Gajski, Prentice Hall, 1997. (教科書)
- The Complexity of Computing, John E. Savage, John Wiley & Sons, 1976. (参考書)
論理回路の複雑さに関する好著．

A-2 コンピュータアーキテクチャ特論 [1]

目的

学部科目の「コンピュータアーキテクチャ」に引続き、各種のコンピュータアーキテクチャの実例について詳細に講述する。特に、コンピュータアーキテクチャ設計の際のハードウェアとソフトウェアのトレードオフの実際例を詳しく比較・対照する。

概要

最新の種々のコンピュータアーキテクチャについて比較・考察し、それらの評価方法について追究する。

[修得項目]

アーキテクチャの分類指標，性能，性能評価手法，コンピュータシステムの評価

[知識項目]

トレードオフ論争，定量的評価，コンピュータアーキテクチャの評価，計算モデルとアーキテクチャ設計，コードスケジューリング，プロセス/スレッド管理，命令レベル並列処理，分岐予測と投機的実行，メモリアクセスの高速化，メモリインタフェースの高速化，マイクロプロセッサ，RISC，スーパーコンピュータ

[関連項目]

VLSI 設計，CAD，分散処理，並列処理

講義項目

1. コンピュータアーキテクチャの比較・分類 [2]

アーキテクチャの分類指標，トレードオフ論争（RISC 対 CISC 論争など），ハードウェア構成方式，汎用コンピュータと専用コンピュータ

2. コンピュータシステムの評価 [2]

性能，性能評価手法，コンピュータシステムの評価とアーキテクチャの評価，定量的評価

3. システムプログラムとコンピュータアーキテクチャ [2]

計算モデルとアーキテクチャ設計，コードスケジューリングにおけるコンパイラとハードウェアのトレードオフ，プロセス/スレッド管理とアーキテクチャ

4. プロセッサアーキテクチャ [3]

演算機能と VLSI 化，命令レベル並列処理（スーパスカラ，VLIW，マルチスレッドなどのアーキテクチャ），分岐予測と投機的実行，プログラム可能ハードウェア

5. メモリアーキテクチャ [2]

メモリアクセスの高速化，メモリインタフェースの高速化，機能メモリ，メモリ保護，種々のメモリ階層

6. 実際のコンピュータアーキテクチャ [4]

マイクロプロセッサ, RISC, マイクロプロセッサを用いた汎用コンピュータ(メインフレーム, ワークステーション, パソコン), スーパーコンピュータ(ベクトルコンピュータ, 並列コンピュータ), 分散処理システム, 非ノイマン型コンピュータ, 高級言語マシン

先修科目

コンピュータアーキテクチャ(学部)

後修科目

なし

関連科目

論理設計と CAD, ハードウェアアルゴリズム論, VLSI システム設計, 分散処理システム, 並列処理

実験及び演習

- ASIC の設計: 特定用途向きの LSI(ASIC) を設計する。他の大学院科目(ハードウェアアルゴリズム, 論理設計と CAD など)と連動して, 設計だけではなく, LSI の試作にまで至ると効果的である。その際は, 大規模集積回路設計教育研究センター(VDEC)などを活用する。

実施方法

- 学部科目の「コンピュータアーキテクチャ」と併せて, 各機関の実現可能性に合わせて学部項目と大学院項目とを切り分け直しても良い。

教科書/参考書

- コンピュータ・アーキテクチャ - 設計・実現・評価の定量的アプローチ - , J.L.Hennessy, D.A.Patterson (富田 眞治・村上 和彰・新實 治男 訳), 日経 BP 社, 1992 (参考書)
コンピュータアーキテクチャの設計, 実現(実装), 評価の各観点から実際的かつ定量的データを提示して詳解している。これを全面改訂した第2版が1996年に出版されている(訳書はまだない)。
- コンピュータアーキテクチャI, 富田 眞治, 丸善, 1994 (参考書)
演算装置, メモリ装置, 制御装置のハードウェア構成について詳解している。また, 実際的な処理方式として並列処理を取り上げ詳説している。
- コンピュータアーキテクチャ, 柴山 潔, オーム社, 1997 (参考書)
マシン命令セット, プロセッサ, メモリ, さらにはマイクロプロセッサやスーパーコンピュータの実際例の各機能ごとに, ハードウェアとソフトウェアとのトレードオフの対照を切口に種々の方式について詳解している。また, システムプログラムとハードウェアとの関連についても詳しい。
- マイクロプロセッサと RISC, 金田 悠紀夫, オーム社, 1991 (参考書)
マイクロプロセッサ, 特に RISC について, 実際例を紹介している。
- コンピュータ・アーキテクチャと RISC, 奥川 峻史, 共立出版, 1992 (参考書)
RISC アーキテクチャを中心に述べている。代表的な RISC の実例が詳解してある。

- スーパーコンピュータ, 長島 重夫・田中 義一, オーム社, 1992 (参考書)
ベクトルコンピュータを中心に述べている。
- データ駆動型並列計算機, 弓場 敏嗣・山口 喜教, オーム社, 1993 (参考書)
「データフローマシン」という並列コンピュータについて詳述している。
- スーパスカラ・プロセッサ - マイクロプロセッサ設計における定量的アプローチ - , M.Johnson(村上 和彰 監訳), 日経 BP 出版センター, 1994 (参考書)
マイクロプロセッサのスーパースカラアーキテクチャについて詳解している。
- 並列計算機アーキテクチャ, 奥川 峻史, コロナ社, 1991 (参考書)
並列コンピュータについて詳解してある。相互結合網については, 特に詳しい。
- 並列コンピュータ, 天野 英晴, 昭晃堂, 1996 (参考書)
並列コンピュータアーキテクチャをメモリアーキテクチャの観点から分類し, 詳説している。
- 並列コンピュータ工学, 富田 眞治, 昭晃堂, 1996 (参考書)
並列コンピュータのハードウェア構成方式について実例を中心に詳解している。また, 並列コンピュータの実際的な応用例にも言及している。

A-3 オペレーティングシステム特論 [1]

目的

オペレーティングシステム(OS)が、新しいコンピュータシステムを社会に浸透させる大きな原動力の1つであることを認識させ、具体的には、以下を目的とする。

- 新しいコンピュータ環境(分散/並列システム, マルチメディアシステム, 組み込みシステム)を対象としたOSの役割を理解させ、設計できる能力を養成すること。
- システムアーキテクチャのあるべき姿について考えさせ、新しいOSを設計できる能力を養成すること。

概要

新しいコンピュータシステムの目的と、その目的に沿ったOSについて、その果たすべき役割と機能、実現方式について講述する。さらに、将来社会における、コンピュータ環境およびOSが果たすべき役割について考察する。

[修得項目]

(大項目)

新しいコンピュータシステムとOSの役割および機能, OS構成法, 新しいOSを考える能力, ユーザインタフェース

(小項目)

分散/並列システム: ハードウェア構成, 各種透過性, カーネルデータの統一性, 各種高速化技法, 軽い実行環境(各種スレッドモデル), 協調スケジューリング, 負荷バランス, プロセス移送, 分散共有メモリシステム, コヒーレンス制御, 同期の種類と高速化手法, 分散/並列ファイルシステム, 入出力システム

マルチメディアシステム: ハードウェア構成, リアルタイム処理, 周期処理, QoS, デッドライン制御

組み込みシステム: 特定用途, ハードウェア構成, 最適化

[知識項目]

分散/並列システム: メモリコンシステンシモデル

マルチメディア: データ圧縮技法, 優先度逆転

組み込みシステム: 開発環境

[関連項目]

なし

講義項目

1. 近代システム概論 [1]

分散/並列システム, マルチメディアシステム, 組み込みシステム, 新しいOSの概論, グラフィカルユーザインタフェース

2. OSの構成法 [1]

基本的手法(モジュール化, 階層化), 単層カーネル, マイクロカーネル, システムサーバ

3. 分散 OS[4]

ハードウェア構成，各種透過性，通信と同期（遠隔手続き呼び出し，通信プロトコル），プロセス管理とスケジューリング（スレッド，負荷バランス，プロセス移送），分散ファイルシステム，メモリコンシステンシモデル，分散共有メモリシステム，事例

4. 並列 OS[4]

ハードウェア構成，カーネルデータの統一性，OS 実現方法（マスタ/スレーブ，リエントラント），各種スレッドモデル（カーネルレベル，ユーザレベル，協調型スレッド），協調スケジューリング，同期の種類とアルゴリズム，メモリ管理（高速化技法，コヒーレンス制御），並列ファイルシステム，並列入出力システム

5. マルチメディア用 OS[3]

ハードウェア構成，データ圧縮技法，リアルタイム処理，周期処理，QoS(Quality of Service)，優先度逆転問題，デッドライン制御，事例

6. 組み込み用 OS[2]

特定用途，ハードウェア構成，各種最適化（カーネルサイズの極小化，消費電力の極小化など），開発環境

先修科目

オペレーティングシステム，情報ネットワーク，コンピュータアーキテクチャ特論

後修科目

なし

関連科目

ヒューマン - コンピュータ・インタラクション，ソフトウェア工学，ソフトウェア工学特論，ネットワークアーキテクチャ，情報セキュリティ，分散処理システム，コンピュータグラフィックス特論

実験及び演習

- 新しい OS の作成：小さなシステムでよいから，新しい OS を作成させたい。

実施方法

- マルチメディアシステム，組み込みシステムについては，適当な参考書が見あたらないので，企業の方に講演して頂くのも，一つの方法である。

教科書/参考書

（一般的）

- Modern Operating Systems, A.S.Tanenbaum, Prentice-Hall, 1992 (参考書) (OS の基礎と応用 - 設計から実装, DOS から分散 OS Amoeba まで, 引地信之 他訳, トッパン, 1995)
古典的な OS から近代の OS (主に分散 OS) までを網羅している。

(並列システム)

- 共有記憶型並列システムの実際，鈴木 則久・清水 茂則・山内 長承，コロナ社，1993(参考書)
共有メモリ型並列システムについて著者の開発マシンの紹介をまじえながら，ハードウェアから応用まで幅広く記述している。
- 並列オペレーティングシステム，福田 晃，コロナ社，1997(参考書)
共有メモリ型並列システムの OS について，基本概念とその体系化，および課題などを記述している。
- UNIX Systems for Modern Architectures – Symmetric Multiprocessing and Caching for Kernel Programmers, C.Schimmel, Addison Wesley, 1994(参考書)(UNIX カーネル内部解析 – キャッシュとマルチプロセッサの管理 – ，前川守 監訳，ソフトバンク，1996)
共有メモリ型並列システムを対象とした UNIX について，実装方法などを詳しく記述している。

(分散システム)

- 分散オペレーティングシステム - UNIX の次にくるもの，前川 守・所 真理雄・清水 謙多郎 編，共立出版，1991(参考書)
分散 OS の基本概念および各種技術と実現方式などを要領よくまとめている。また，有名な分散 OS についての解説もある。
- Distributed Operating System, A.S.Tanenbaum, Prentice-Hall, 1995(参考書)(分散オペレーティングシステム，水野 忠則 他 訳，トッパン，1996)
分散 OS の概念と各種記述など全般にわたり要領よく記述している。分散共有メモリシステム，メモリコンシステンシモデルを記述しているところに特徴がある。
- Distributed Operating Systems - Concepts and Design, P.K.Sinha, IEEE Computer Society Press, 1997(参考書).
分散 OS の基本概念と設計原理を要領よく記述している。

A-4 データベース特論 [1]

目的

高度応用のために提案されているさまざまなデータモデルの基礎概念とそれぞれの基本的な手法を理解させ、データモデルとデータベースシステムの進化について考えさせる。

概要

関係データモデルの長所と短所をまとめ、演繹データベース、オブジェクト指向データベース、演繹オブジェクト指向データベース、データベースプログラミング言語など種々の拡張について学習する。

[修得項目]

関係データベースの長所と短所、演繹データベース、オブジェクト指向データベース、演繹オブジェクト指向データベース、データベースプログラミング言語

[知識項目]

データベースの形式的意味論、問合せ最適化(上昇評価と下降評価の技術)、Datalog、オブジェクト指向の諸概念、オブジェクトの形式的表現と意味論、構造表現、インピーダンス不整合の克服法

[関連項目]

論理プログラミング、定理証明、プログラミング言語における形式的意味論、知識情報処理、非単調推論、型推論、反基礎公理集合論

講義項目

1. データベース進化論 [2]

背景、関係、演繹、オブジェクト指向、演繹オブジェクト指向、データベースプログラミング、意味、関数

2. 演繹データベース [3]

論理の基礎、関係の論理的再構築、Datalog、一貫性制約

3. 演繹データベースの問合せ処理 [3]

最適化、上昇評価、下降評価、さまざまな最適化手法

4. オブジェクト指向データベース [3]

オブジェクト指向の諸概念、マニフェストの評価、マルチメディア応用、さまざまな実システム

5. 演繹オブジェクト指向データベース [3]

論理と構造の干渉、オブジェクト指向諸概念の論理的表現、構造の表現、さまざまな言語/システム

6. データベースプログラミング言語 [2]

型システム、型推論、種々のプログラミングパラダイムに基づいた言語

先修科目

データベース，プログラミング言語論，数理論理学，ソフトウェア工学，人工知能，コンパイラ

後修科目

なし

関連科目

プログラム意味論, ヒューマン - コンピュータ・インタラクション

実施方法

- すべての項目を行なうと2半期必要であるが，もし，半期で本科目を実施するならば，1，2の全項目と3，5の一部を最低限学習するのが望ましい．

教科書/参考書

- 新データベース論 — 関係から演繹オブジェクト指向へ，横田・宮崎，共立出版，1994. (教科書)
データベース進化論の立場から種々のデータモデルが述べられており，この科目の目標に適した教科書である．
- Principles of Database and Knowledge-base Systems, Ullman, Computer Science Press, 1988. (参考書)
幅広いトピックの原理と関連が述べられており，上記教科書を補うのに最適である．

備考 (コメント)

- 論理プログラミングやプログラミング言語の形式的意味論の授業が他であれば，内容の調整が必要．もしなければこのためにある程度の時間が必要となる．

A-5 画像情報処理特論 [1]

目的

人間の視覚認識機能の工学的な実現を目的とした画像の認識処理について講述する。特に、3次元シーンの認識を目指すコンピュータビジョンの基礎知識を修得させる。単に手法を教えるのではなく、コンピュータビジョンにおける問題の難しさが何に起因しているのかを理解させるとともに、問題を数学的に定式化し、その解法を探るといった方法論の重要性を理解させる。

概要

コンピュータビジョンの基礎知識を修得する。2次元画像からシーンの3次元構造を推定するための基本的なアプローチを学ぶとともに、画像認識の基礎となる特徴の表現方法と識別・認識手法の概要を理解する。

[修得項目]

射影モデル, 画像特徴とシーン特徴の関係, 画像化の逆問題, 不良設定問題, 標準正則化理論, エネルギー最小化, 弛緩法, 両眼ステレオ, 特徴空間表現, 部分空間法, 関係構造グラフ, 統計的パターン識別, クラスタリング, テンプレートマッチング

[知識項目]

視覚の計算理論, 動的輪郭モデル, Shape-from-X (Texture, Shading, Motion, Contour), 曲率解析, 拡張ガウス像, グラフマッチング, DP マッチング, 制約フィルタリング

[関連項目]

カラー反射モデル, 能動的距離計測, レーザレンジファインダ, 距離画像, 照度差ステレオ, アスペクトグラフ, ボリューム表現, 一般化円筒, CSG 表現, 関数表現, 環境モデル, ニューラルネットワーク, 遺伝的アルゴリズム, シミュレーテッドアニミング, 自律移動ロボット, 人工現実感

講義項目

1. 画像とシーン [2]

射影モデル, カラー反射モデル, 視覚の計算理論, 画像特徴とシーン特徴の関係, 画像化の逆問題

2. 正則化手法 [2]

不良設定問題, 標準正則化理論, エネルギー最小化, 弛緩法, 動的輪郭モデル

3. 3次元情報の復元 [4]

両眼ステレオ, Shape-from-X (Texture, Shading, Motion, Contour), 能動的距離計測, レーザレンジファインダ, 距離画像, 照度差ステレオ

4. 特徴表現 [3]

特徴空間表現, 部分空間法, 関係構造グラフ表現, アスペクトグラフ, 曲率解析, 拡張ガウス像, ボリューム表現, 一般化円筒, CSG 表現, 関数表現, 環境モデル

5. 識別・認識手法 [4]

統計的パターン識別, クラスタリング, テンプレートマッチング, グラフマッチング, DP マッチング, 制約フィルタリング, ニューラルネットワーク, 遺伝的アルゴリズム, シミュレーテッドアニーリング

先修科目

確率論と情報理論, 数理計画法, 数値計算, パターン認識, 信号処理, 画像情報処理

後修科目

なし

関連科目

デジタル通信, コンピュータグラフィックス, コンピュータグラフィックス特論, ヒューマン-コンピュータ・インタラクション, 信号処理特論, 音声情報処理, 人工知能, 集積回路工学

実験及び演習

- 画像処理実験: 画像処理ソフトウェアパッケージや DSP ボードを用いた実験及び演習を連動させるとよい。

実施方法

- 講義項目と関連づけて具体的な応用事例や最近の研究のトピックを紹介するとよい。
- 項目 5 の後半については全てをカバーする必要はなく, トピックを選択すればよいと思われる。

教科書/参考書

- The Psychology of Computer Vision, P.H. Winston, Ed., McGraw-Hill, 1975 (参考書)
コンピュータビジョンの心理, 白井良明, 杉原厚吉 訳, 産業図書, 1979
1970 年代前半までに MIT 人工知能研究所で行われた研究のトピックを紹介している。
- Vision — A Computational Investigation into the Human Representation and Processing of Visual Information —, D. Marr, Freeman, 1982 (参考書)
ビジョン — 視覚の計算理論と脳内表現 —, 乾敏郎, 安藤広志 訳, 産業図書, 1987
1980 年代のコンピュータビジョン研究に大きな影響を与えた著者の視覚の計算理論がまとめられている。
- Computer Vision, D.H. Ballard and C.M. Brown, Prentice-Hall, 1982 (教科書)
コンピュータ・ビジョン, 福村晃夫 監訳, 日本コンピュータ協会, 1987
コンピュータビジョンの基本アルゴリズムが体系的かつ丁寧に述べられており, 数学的な道具立ても示されている。
- Robot Vision, B.K.P. Horn, McGraw-Hill, 1986 (参考書)
ロボットビジョン — 機械は世界をどう見るか —, NTT ヒューマンインタフェース研究所プロジェクト RVT 訳, 朝倉書店, 1993
3次元シーンの撮像から認識に至る各過程のアルゴリズムが体系的にまとめられている。演習問題も豊富である。

- Encyclopedia of Artificial Intelligence, S.C. Shapiro Ed., John Wiley & Sons, 1987 (教員用参考書)
人工知能大辞典, 大須賀節雄 監訳, 丸善, 1991
人工知能全般に関する辞典であり, コンピュータビジョンの代表的な分野・アプローチについて歴史的な研究の流れを含めて記述されている.
- マシンビジョン, 江尻正員, 大田友一, 池内克史 共著, 昭晃堂, 1990 (参考書)
コンピュータビジョンの基本手法と産業応用の事例を紹介している.
- 三次元画像計測, 井口征士, 佐藤宏介 共著, 昭晃堂, 1990 (参考書)
3次元計測手法についてまとめたもので, 特に, 能動的距離計測に関して詳しく述べられている.
- 画像理解 — 3次元認識の数理 —, 金谷健一, 森北出版, 1990 (参考書)
動画像解析における基本的な問題を射影幾何学, テンソル解析の視点から統一的に記述している.

備考 (コメント)

- 方法論と基本手法の体系的な理解に主眼をおいた講義構成になっているが, 産業応用の事例や最近の映像メディア処理, 拡張現実感への応用をトピック的に示すことによって応用イメージを理解させることも重要である.

A-6 コンピュータグラフィックス特論 [1]

目的

コンピュータグラフィックスのアルゴリズムを効率的に実現するシステムや動きの表示法，アニメーションの作成，実写映像との合成法などについて理解させ，よりリアルな画像を生成するための問題について考えさせる．

概要

コンピュータグラフィックスで学習した基礎知識をもとに，アルゴリズムの高速化，高機能化，自然な動きの表現方法，実写画像との合成方法などについて学習する．

講義項目

1. アドバンスドグラフィックス [3]

非線形幾何変換，ラスタ変換，3D モデリング技術，高度なレンダリング技術，実映像との合成法

2. グラフィックスアーキテクチャ [3]

ラスタ型グラフィックス，ビデオプロセッサ，並列処理

3. アニメーション [3]

アニメーションの基礎，動きの表現法，アニメーション記述言語衝突計算法

4. 仮想空間の生成と対話 [3]

3D 空間の表示法，実時間対話処理，マルチモーダルインターフェース，画像認識処理との統合

先修科目

コンピュータグラフィックス，画像情報処理，ヒューマン - コンピュータ・インタラクション，音声情報処理

後修科目

なし

関連科目

画像情報処理特論（大学院），音声情報処理（大学院）

実験及び演習

- 必要なし

実施方法

- どれかの項目を一つ選んで，最新の研究成果についての話題と関連つけて説明してもよい．

教科書/参考書

大学院では論文を中心に講義を進めるのが望ましい。また、この分野は映画や芸術関連の出版物にも教科書となり得るものもある。

- Computer Graphics principles and practice, Foley et al., Addison Wesley, 1992 (教科書)
原理と応用が述べられており、一部の話題には適した教科書である。

備考 (コメント)

- できれば「画像情報処理特論」とは別科目としたいが、やむを得ない場合は、同一科目とすることも可能である。

A-7 信号処理特論 [1]

目的

実世界で観測される信号を現実のハードウェアデバイスを用いていかに信号処理するかについて講述する。単にデジタル信号処理の方法を学ぶのではなく、与えられた信号処理の要求から、実際に信号処理ハードウェアの基本設計が行えるようになることを目指す。さらに、具体的な音響信号や画像信号を処理させることで、実際の音響・映像装置やモデム装置などの内部で行われている信号処理を理解させる。

概要

デジタル信号処理の基本となるデジタルフィルタの概要と設計法を学ぶとともに、信号処理用デバイスやハードウェアの基礎知識についても修得する。さらに、2次元信号に対しフィルタリングを行うことで各種画像処理手法を体得する。

[修得項目]

デジタルフィルタの数学的表現 (伝達関数表現, 状態空間表現), インパルス応答, 周波数フィルタ, FIR フィルタの周波数領域設計, FIR フィルタの時間領域設計, IIR フィルタの周波数領域設計, IIR フィルタの時間領域設計, デジタルフィルタの構成要素, デジタルフィルタ標準形, デジタル信号プロセッサ (DSP), アナログインタフェース 2次元離散フーリエ変換, 2次元デジタルフィルタ,

[知識項目]

Z変換, システム応答, デルタ関数ハイパスフィルタ, ローパスフィルタ, バンドパスフィルタ, 移動平均, パタワースフィルタ, チェビシェフフィルタ, 畳み込み積分, 畳み込み演算器 (コンボルバ), コンボリューションカーネル, フィルタ次数, タップ数, 因果律, 信号遅延, 窓関数, A/D変換器, D/A変換器, ハーバードアーキテクチャ, 積和演算回路

[関連項目]

画像圧縮, モデム回路, アナログフィルタ

講義項目

1. デジタルフィルタの基礎 [2]

システム応答, 周波数フィルタ (ハイパス, ローパス, バンドパス), フィルタの数学的表現 (伝達関数表現, 状態空間表現), インパルス応答,

2. 有限応答デジタルフィルタ (FIR フィルタ) の設計 [3]

フィルタ構造, 畳み込み積分 (コンボリューション) とカーネル, 周波数領域設計, 時間領域設計, 因果律, 窓関数, フィルタ次数

3. 無限応答デジタルフィルタ (IIR フィルタ) の設計 [3]

フィルタ構造, 標準形, 周波数領域設計, 時間領域設計, パタワースフィルタ, チェビシェフフィルタ

4. デジタル信号処理のハードウェア [2]

フィルタ回路構成要素 (乗算器, 加算器, メモリ), 積和演算回路, タップ数, アナログフィルタ対デジタルフィルタ

5. アナログインタフェース [1]

A/D変換器, D/A変換器

6. デジタル信号プロセッサ [2]

デジタル信号プロセッサ (DSP) の歴史, DSP アーキテクチャ, DSP 応用

7. 2次元デジタル信号処理 [2]

2次元離散フーリエ変換, 2次元デジタルフィルタ, 画像処理応用

先修科目

解析学, 複素関数論, 信号処理, 論理設計

後修科目

画像情報処理特論, 音声情報処理

関連科目

制御理論, システム理論, 画像情報処理, 音響処理, デジタル通信, 集積回路工学

実験及び演習

デジタルフィルタを実際に設計させ, その性能を体得させる.

- FIR フィルタ: FIR フィルタを実際に設計させ, 与えた一次元信号から正しくフィルタリングできていることを修得させる. またタップ数の変化で, フィルタ性能が異なることを確認させる.
- IIR フィルタ: IIR フィルタを実際に設計させ, FIR フィルタと同じ入力信号から, FIR フィルタよりシャープにフィルタリングできていることを確認させる.
- 2次元信号処理: 2次元信号として画像データを与え, それをフィルタリングすることで画像鮮鋭化やノイズ除去などの各種画像処理が実現できることを修得させる.

実施方法

- 講義時間内に実験や演習を行うと, 時間が足りなくなる恐れがある. その場合には, IIR フィルタと2次元信号処理の演習を省略し, FIR フィルタの設計演習のみとする.

教科書/参考書

- Digital Signal Processing, W. Mayeda 著, Prentice-Hall, 1993 (参考書)
- デジタル信号処理ハンドブック, 電子情報通信学会編, 電子情報通信学会, 1993 (教員用参考書)
- デジタルフィルタ設計入門, 尾知博 著, CQ出版, 1990 (教員用参考書)
- DSP 活用のポイント, 石田義久, 鎌田弘之 共著, 産業図書, 1990 (教員用参考書)
- 信号・画像のデジタル処理, 有本卓 著, 産業出版, 1980 (教員用参考書)

- デジタルフィルタデザイン, 三谷政昭 著, 昭晃堂, 1987 (教科書)
上級者向き.
- シグナルプロセッサとその応用, 丸田力男, 西谷隆夫 共著, 昭晃堂, 1988 (教科書)
上級者向き.
- ビギナーズデジタルフィルタ, 中村尚五 著, 東京電機大学出版局, 1989 (教科書)
初心者向き.
- CAI デジタル信号処理, 小畑秀文, 幹康 共著, コロナ社, 1991 (参考書)
初心者向き.

A-8 ソフトウェア工学特論 [1]

目的

学生演習や研究室におけるプログラムの作成と、産業としてのソフトウェアの開発とは、全く異なることを認識させて、問題意識を持たせる。ソフトウェア開発における問題点と、それに対する解決法ならびにそれらの特徴を理解させる。

概要

工業的生産活動としてのソフトウェア開発におけるプロセスとその管理について述べ、そこでの問題点、実施および提案されている解決法、ならびにそれらの限界などを議論する。また、ソフトウェア開発に関して最近話題になっている事柄の紹介も行う。

[修得項目]

コスト意識、インタフェース(モジュール間、開発プロセスの段階間、発注者と受注者間、ユーザと開発者間、開発担当者間、等々)の重要性、ソフトウェアプロセスモデル

[知識項目]

定量化・見積り・管理(各手法の特長と限界)

[関連項目]

サイエンス/テクノロジー/エンジニアリング、ソフトウェア産業

講義項目

1. ソフトウェアプロセス [1]

ウォーターフォールモデル、螺旋モデル、プロトタイピング

2. 要求定義 [2]

問題分析、システムモデル、要求獲得、要求の妥当性、要求仕様記述、要求の進化

3. 保守 [3]

部品化、再利用、バージョン管理、構成管理

4. 管理 [3]

コスト見積り、進捗管理、品質管理、定量化、組織運営、コミュニケーション、情報共有、プロセス改善

5. ツールと環境 [3]

CASE ツール、統合環境、グループウェア、ユーザインタフェース

6. トピックス [3]

形式的手法、コンポーネントウェア、エンドユーザプログラミング、ドメインエンジニアリング、高信頼性システム、実時間処理システム、組み込みシステム、ISO9000、ISO14000、CALIS、著作権、知的所有権

先修科目

ソフトウェア工学

後修科目

なし

関連科目

データベース, ヒューマン - コンピュータ・インタラクション

実験及び演習

- ソフトウェアシステム開発実習
学外実習等で, ソフトウェアシステムの開発現場に立ち会って実際上の問題に触れる機会を学生に与える.

実施方法

- 企業等でソフトウェア開発の現場にいる技術者を講師に招いて実際上の問題点とその解決法について話して貰うことができると良い.
- 大企業などで持っている社内の研修コースの一部を利用できたり, 参考にできたりすると, 有益であろう.

教科書/参考書

- 松本吉弘: ソフトウェア工学, 丸善, 1992. (教科書)
ソフトウェア工学全般に関して理論的な背景も含めて要領良くまとまった良書である.
- Frederick P. Brooks, Jr. : The Mythical Man-Month - Essays on Software Engineering, Anniversary Edition, Addison-Wesley, 1995.
滝沢、牧野、富澤訳: 人月の神話 - 狼人間を撃つ銀の弾はない -, 星雲社, 1996. (参考書)
ソフトウェア開発の実際を述べた歴史的な名著. 副読本としても好適.
- Ian Sommerville : Software Engineering, Fifth Edition, Addison-Wesley, 1996. (参考書)
米国での定評ある教科書である.
- Caper Jones : Assessment and Control of Software Risks, Prentice Hall, 1993.
島崎・富野(監訳): ソフトウェア病理学 - システム開発・保守の手引, 共立出版, 1995. (参考書)
ユニークな構成の本であるが, ソフトウェア開発における問題点を項目毎に網羅的に解説しており, 事典的な参照にも便利である.

大学院科目 **G**

ハードウェアアルゴリズム論 **G-1**

論理設計と CAD **G-2**

プログラム意味論 **G-3**

ネットワークアーキテクチャ **G-4**

分散処理システム **G-5**

情報セキュリティ **G-6**

関数プログラミング **G-7**

オブジェクト指向プログラミング **G-8**

音声情報処理 **G-9**

アルゴリズム論 **G-10**

G-1 ハードウェアアルゴリズム論 [1]

目的

集積回路を計算モデルとした計算論である。計算時間，消費電力，面積などを評価尺度にした各種アルゴリズムの評価を行うとともに，集積回路設計に関する CS 的な理論的指針を確立することを目指す。内容は，担当者の選択に大きく依存する。ここで示すものは 1 例である。

概要

集積回路を計算モデルとした VLSI モデル上でのアルゴリズム設計論を展開する。アルゴリズムの評価尺度としては，チップ面積，計算時間，消費電力などを使う。各種の実用的な演算に対するハードウェアアルゴリズムを解析評価するとともに，基本的なハードウェアアルゴリズム設計法を講義する。

[修得項目]

ハードウェアアルゴリズム，VLSI モデル，ハードウェアアルゴリズムの評価尺度，面積，時間，消費電力，並列計算，パイプライン処理，VLSI 上での複雑さ，面積複雑度，時間複雑度，エネルギー複雑度

[知識項目]

符号化，シストリックアルゴリズム，加減算のアルゴリズム，乗算のアルゴリズム，初等関数のアルゴリズム，行列演算のアルゴリズム，ソーティングのアルゴリズム，記号処理のアルゴリズム，図形処理と画像処理のアルゴリズム，組み合わせ問題のアルゴリズム

[関連項目]

論理回路，並列計算，アルゴリズム論，計算の複雑さ

講義項目

1. ハードウェアアルゴリズムとは [2]

ハードウェアアルゴリズムの定義，関連分野との関係

2. VLSI モデルとアルゴリズム解析 [2]

VLSI モデル，ハードウェアアルゴリズムの評価尺度，面積，時間，消費電力

3. ハードウェアアルゴリズムの設計手法 [3]

並列計算，パイプライン処理，符号化，シストリックアルゴリズム

4. 複雑さの理論 [2]

VLSI 上での複雑さ，面積複雑度，時間複雑度，エネルギー複雑度，複雑度間のトレードオフ

5. 算術演算のハードウェアアルゴリズム [2]

加減算のアルゴリズム，乗算のアルゴリズム，初等関数のアルゴリズム，行列演算のアルゴリズム

6. 非数値演算のハードウェアアルゴリズム [2]

ソーティングのアルゴリズム，記号処理のアルゴリズム，図形処理と画像処理のアルゴリズム，組み合わせ問題のアルゴリズム

教科書/参考書

- Computational Aspects of VLSI, Jeffrey D. Ullman, Computer Science Press, 1984. (日本語訳: VLSI 計算諸側面, 都倉ほか訳, 近代科学社, 1990.) (参考書)
VLSI を計算モデルとした計算の複雑さの理論を解説している.
- Introduction to VLSI systems, Carver Mead and Lynn Conway, Addison Wesley, 1980. (教科書)
システム設計の立場から VLSI をどのように設計するかを議論し, 時代を変えた有名な指導書.

G-2 論理設計と CAD[1]

目的

論理設計に関係した CAD 技術の講義を紹介する。論理合成，レイアウト合成，テスト生成など多くの最適化問題に対する最適化アルゴリズムとそのベースとなった考え方を講義する。VDEC を利用して，実際のツールを利用した実習および設計実験を並行して行うことが望ましい。内容は，担当者の選択に大きく依存する。

概要

論理回路の設計を支援する CAD 技術の現状を紹介する。論理合成，形式的論理検証，レイアウト合成，テスト生成，シミュレーションなどの要素技術の原理と，それらのベースとなる最適化手法について解説する。さらに，今後重要となるとと思われるアーキテクチャ設計のための CAD 技術やハードウェア - ソフトウェア協調設計の技術についても触れる。

[修得項目]

配置と配線，2 段最適化，多段化，テクノロジーマッピング，トランスダクション法，論理シミュレーション，テスト生成法，故障シミュレーション，高位合成手法，アーキテクチャ設計支援手法，

[知識項目]

CMOS 技術，スタンダードセル方式，ゲートアレイ方式，フルカスタム方式，形式的検証法，2 分決定図の利用，回路のテスト，テスト容易化設計，VHDL，Verilog HDL，UDL/I，ハードウェアとソフトウェアの分割，システムシミュレーション，仕様記述，ソフトウェア設計支援技術

[関連項目]

デジタル集積回路設計，最適化手法，組み込みシステム設計

講義項目

1. デジタル集積回路とその設計の流れ [1]

論理回路設計と CAD ツール

2. 回路の実現技術と設計手法 [1]

CMOS 技術，スタンダードセル方式，ゲートアレイ方式，フルカスタム方式

3. レイアウト手法とレイアウト合成ツール [2]

配置と配線，最適化手法，性能優先と面積優先

4. 論理合成手法 [2]

2 段最適化，多段化，テクノロジーマッピング，トランスダクション法

5. 論理検証法 [2]

論理シミュレーション，形式的検証法，2 分決定図の利用

6. テスト設計とテスト容易化設計法 [2]

回路のテスト，テスト生成法，故障シミュレーション，テスト容易化設計

7. 設計記述言語と高位合成 [1]

VHDL , Verilog HDL , UDL/I , 高位合成手法

8. アーキテクチャ設計とフロアプラン [1]

アーキテクチャ設計支援手法 , フロアプラン設計支援

9. ハードウェア - ソフトウェア協調設計とシステム設計 [1]

システム設計 , ハードウェアとソフトウェアの分割 , システムシミュレーション , 仕様記述 , ソフトウェア設計支援技術

教科書/参考書

- Principles of CMOS VLSI Design – A systems Perspective –, Second Edition, Neil H.E. Weste and Kamran Eshraghian, Addison Wesley, 1993. (教科書)
情報系の学生が知っておくべき集積回路に関する知識が非常にわかりやすく述べられている名著 .
日本語訳 : CMOS VLSI 設計の原理 –システムの視点から– , 富沢 , 松山監訳 , 丸善 , 1988. (ただし , 第 1 版の訳である) .
- CMOS 集積回路 - 入門から実用まで - , 榎本忠儀 , 培風館 , 1996 (教科書)
レイアウトの実習まで行うときには実用的な教科書 .
- Mordern VLSI Design -A System Approach -, Wayne Wolf, Printice Hall, 1996. (教科書)
システム設計者の立場からの VLSI 設計論 .
- 集積回路設計入門 , 国枝博昭 , コロナ社 , 1996 (教科書)
- CMOS Digital Integrated Circuits - Analysis and Design -, Sung-Mo Kang and Yusuf Leblebici, McGraw Hill, 1996. (参考書)
- ULSI の効果的な設計法 , 中村行宏 小野定康 , オーム社 , 1994. (参考書)
Parthenon を利用して大規模集積回路の設計を進める過程を解説している .

G-3 プログラム意味論 [1]

目的

プログラムの意味論は、プログラムの検証法とともに発達してきたが、単なる検証だけではなく、仕様の形式的な記述法やプログラミング方法論にまで大きな影響を与えている。また、数学的にも興味深い話題が多い。本科目では、代表的なプログラムの意味論を教えるとともに、プログラムの検証法や、プログラムの形式的な取り扱いなどを修得させる。

概要

プログラミング言語の様々な意味定義手法について学び、プログラムの形式的な取り扱いやプログラムの検証などを修得する。

[修得項目]

プログラムの検証方法

[知識項目]

操作的意味論，表示の意味論，公理の意味論，ドメイン理論

[関連項目]

形式的仕様記述，並行計算の記述・意味論

講義項目

1. プログラムの意味論への導入

プログラムの検証とは何か。意味論がなぜ必要か。基本的な概念の準備

2. 操作的意味論 [3]

式の評価，コマンドの実行，種々の帰納的定義と帰納法原理，証明法

3. 公理の意味論 [3]

Hoare 規則，健全性と完全性，プログラムの検証

4. 表示の意味論 [3]

数学的な準備，意味関数の定義方法，ドメイン理論

5. 形式的仕様記述，並行計算の記述 [1]

各種仕様記述，CCS, CSP など

先修科目

離散数学，計算論，プログラミング言語論，ソフトウェア工学

教科書/参考書

- The Formal Semantics of Programming Languages, Glynn and Winskel, MIT Press, 1993 (教科書)
本科目の内容に即した構成になっている。

G-4 ネットワークアーキテクチャ[1]

目的

コンピュータネットワークに関する最新的话题を中心に，コンピュータネットワークを設計/構築するための基礎技術および応用技術を明らかにする．また，そのために必要となる性能評価手法を示す．まず，性能評価を行なうための基礎理論となる待ち行列理論/待ち行列網理論を導入し，基本的な M/M/1 待ち行列，M/G/1 待ち行列などの解析手法を示す．さらに，その適用例として，コンピュータネットワークシステムをモデル化し，そのモデルを対象とした性能評価例を述べる．特に講義を通して，ネットワーク性能とコストの間に存在するトレードオフ関係を理解させ，自分でコンピュータネットワーク（の一部）を設計するセンスを身につけさせることを目的とする．

概要

コンピュータネットワークに関する最新的话题を中心に，コンピュータネットワークを設計/構築するための基礎技術および応用技術を明らかにする．また，そのために必要となる性能評価手法を示す．まず，性能評価を行なうための基礎理論となる待ち行列理論/待ち行列網理論を導入し，基本的な M/M/1 待ち行列，M/G/1 待ち行列などの解析手法を示す．さらに，その適用例として，コンピュータネットワークシステムをモデル化し，そのモデルを対象とした性能評価例を述べる．

[修得項目]

M/M/1，M/G/1，Network of Queues，マルチアクセスプロトコル（ALOHA，LAN における CSMA/CD，Token 方式），ルーティング制御，ウィンドウフロー制御，レート制御，通信サービス品質，ベストエフォート型ネットワーク，品質保証型ネットワーク

[知識項目]

B-ISDN，ATM，インターネット，ショーテストパズルーティング，リーキーバケット，フロー制御における Go-back-N/Selective Ack/スロースタート，RTP/RSVP などのリアルタイム保証型プロトコル

[関連項目]

オペレーティングシステム，コンピュータアーキテクチャ，画像/音声の符号化圧縮，マルチメディア処理

講義項目

1. 最新ネットワーク技術の紹介 [2]

B-ISDN，ATM，インターネットなど

2. 待ち行列理論とコンピュータネットワークへの応用 [4]

M/M/1，M/G/1，Network of Queues

3. マルチアクセスプロトコル [2]

マルチアクセスプロトコル（ALOHA，LAN における CSMA/CD，Token 方式など）の評価，CSMA/CD とトークンリングの定性的/定量的比較

4. ルーティング制御 [2]

ルーティング方式の紹介，基本アルゴリズム，分散型への適用と問題点，技術の発展と共に「最適な」アルゴリズムは変化することを知る

5. フロー制御 [3]

ウィンドウフロー制御とレート制御の比較評価，ネットワークの高速化に伴う問題点の指摘（事例研究として）TCP の最近の動向

6. マルチメディア制御 [2]

通信サービス品質，ベストエフォート型ネットワークから品質保証型ネットワークへ，RTP，RSVP などリアルタイム保証型プロトコルによる事例紹介

先修科目

コンピュータアーキテクチャ，オペレーティングシステム，情報ネットワーク，デジタル通信

後修科目

分散処理システム

関連科目

画像情報処理，情報セキュリティ

実験及び演習

待ち行列理論の学習の動機付けを行うために，現実の問題に即した演習問題を課すことが特に重要．

- M/M/1，M/M/s などによる比較：サービス統合ネットワークの利点
- M/G/1：ATM の最適セル長
- M/G/1 Priority Queue：音声/データ統合ネットワークにおける最適パケット長の検討
- Networks of Queues による評価：ウィンドウフロー制御におけるウィンドウサイズ

教科書/参考書

- Fundamentals of Telecommunication Networks, Tarek N. Saadawi and Nistafa H. Ammar, Wiley Interscience, 1994. (教科書)
最新の技術動向も反映された好書である
- Computer Networks; Third Edition, Andrew S. Tanenbaum, Prentice-Hall International, 1996 (参考書)
コンピュータネットワークに関する諸知識が幅広く触れられている
- TCP/IP Illustrated, Volume 1: The Protocols, W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (教員用参考書)
- TCP/IP Illustrated, Volume 2: The Implementation, Gray R. Wright and W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (教員用参考書)

- TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP, and UNIX Domain Protocols, W. Richard Stevens, Addison-Wesley Publishing Company, 1994 (教員用参考書)
3冊のうち, 概念を修得するなら第1部のみでよい. また, ソースコードレベルまでの修得を目指すのであれば第2部も必要. 最新の技術を反映したものは第3部に示されている.
- Internetworking with TCP/IP Volume I - Principles, Protocols, and Architecture -, Douglas E. Comer, Prentice-Hall International, 1991 (参考書)
- Internetworking with TCP/IP Volume II - Design, Implementation, and Internals -, Douglas E. Comer and David L. Stevens, Prentice-Hall International, 1994 (教員用参考書)
- Internetworking with TCP/IP Volume III - Client-Server Programming and Applications -, Douglas E. Comer and David L. Stevens, Prentice-Hall International, 1994 (教員用参考書)
以上の3冊は邦訳「TCP/IPによるネットワーク構築」(共立出版)がある. 第1巻はTCP/IPプロトコル体系の概念をまとめたものであり, 第2巻では内部構造を解説している. ネットワークプログラミングを行なうのであれば第3巻も必要であろう.

備考 (コメント)

- 「情報ネットワーク」との違いは, 自分でコンピュータネットワーク(の一部)を設計する能力を身につけさせることである.

G-5 分散処理システム [1]

目的

近年、コンピュータネットワークの発展に伴って複数のコンピュータを結合した分散処理システムの構築が盛んに行なわれている。本講義では、分散システムのためのソフトウェアの基礎理論と実際について取り扱う。具体的には、分散オペレーティングシステムをはじめとする分散システムのための基本ソフトウェアに必要な機能とその実現法、および関連する諸問題について詳説する。また、最近の技術動向/研究動向の紹介も適宜行なっていく。

概要

近年、コンピュータネットワークの発展に伴い複数のコンピュータを結合した分散システムの構築が盛んに行なわれている。本講義では、分散システムのためのソフトウェアの基礎理論と実際について扱う。

[修得項目]

同期通信と非同期通信，プロセス管理の方法，メモリ管理の方法，

[知識項目]

プロセス移動，プロセススケジューリング，マルチスレッド，mach

[関連項目]

マルチキャスト通信，プロトコル仕様記述言語，分散データベース，トランザクションシステム，グループウェア

講義項目

1. 同期通信と非同期通信 [1]

分散処理システムの機能要件，プロセス間同期の例 (dining philosopher, readers/writers problem)

2. 分散オペレーティングシステムとは [1]

分散オペレーティングシステムの紹介

3. プロセス間通信とプロセス管理 [3]

メッセージ通信，プロセス管理，プロセス移動，プロセススケジューリング

4. 軽量プロセス [1]

マルチスレッド

5. 事例紹介 [2]

mach の例の紹介

6. 分散 OS のメモリ管理 [2]

分散共有メモリ，キャッシング

7. マルチキャスト通信 [1]

マルチキャスト通信プロトコルとプロセス間通信

8. 通信プロトコルの仕様記述 [1]

LOTOS などの仕様記述言語の紹介

9. 分散データベース [1]

分散データベースの紹介

10. トランザクションシステム [1]

トランザクションシステムの紹介

11. グループウェア [1]

グループウェアの紹介

先修科目

計算論，離散数学，コンピュータアーキテクチャ，情報ネットワーク，オペレーティングシステム

後修科目

なし

関連科目

情報セキュリティ，ネットワークアーキテクチャ

実験及び演習

- マルチスレッドを利用した並列処理プログラムの実現
分割統治プログラム，クイックソートなどが考えられる．

実施方法

- 分散処理システムは多岐に渡るため，後半のマルチキャスト通信，プロトコル仕様記述言語，分散データベース，トランザクションシステム，グループウェアについては個々の専門分野に応じて，どれかを選んで深く教えた方がよいであろう．

G-6 情報セキュリティ[1]

目的

現代の情報セキュリティ技術には、計算量理論的安全性に基づく技術と、情報理論的安全性に基づく技術がある。その両者について、その代表例を紹介する。本科目は、情報セキュリティの技術的な側面が中心である。社会的な観点から見た情報セキュリティのあり方や課題などについては、別途教える必要がある。

概要

情報セキュリティ技術として最近開発されてきた様々な技術を学ぶ。具体的には、計算量理論的安全性に基づく技術（公開鍵暗号，デジタル署名，零知識証明）と情報理論的安全性に基づく技術（秘密分散共有法，認証コード）について学ぶ。

[修得項目]

情報セキュリティ技術の原理（計算量理論的安全性，情報理論的安全性），情報セキュリティ技術の使い方

[知識項目]

鍵共有暗号，公開鍵暗号，デジタル署名，零知識証明，秘密分散共有法，認証コード，暗号の安全性

[関連項目]

情報セキュリティの社会的な意味と課題，情報倫理

講義項目

1. 公開鍵暗号への導入 [1]

鍵共有暗号，ElGamal 暗号

2. RSA 暗号 [2]

数論の基礎知識，RSA 暗号，デジタル署名，その他の公開鍵暗号

3. 零知識証明 [2]

電話でコイン投げ，零知識証明，零知識証明の応用例

4. 情報理論的安全性に基づく技術 [3]

秘密鍵分散共有法，認証コード

5. 様々な暗号解読に対する安全性 [2]

DES に対する攻撃，RSA に対する攻撃

先修科目

離散数学，情報代数と符号理論，確率論と情報理論

関連科目

アルゴリズム論

教科書/参考書

- 暗号理論の基礎, D. Stinson 著, 櫻井 幸一 監訳, 共立出版, 1995 (教科書)
この科目の講義内容をほぼ網羅している.
- 情報セキュリティの科学, 太田 和夫, 黒澤 馨, 渡辺 治 著, 講談社, 1995 (参考書)
情報セキュリティ技術の原理と応用例を, わかりやすく解説している. 副読本として推薦するとよいだろう.

G-7 関数プログラミング [1]

目的

関数 (型) プログラミングは実用的側面と、理論的側面を合わせ持つ興味深い分野である。本科目では、その両面に渡って基礎的知識だけでなく、実際のプログラミングを実習を通して習得させる。

概要

関数 (型) プログラミングについて、基礎であるラムダ計算と、LISP という実用的言語を例として基本的事項を学ぶ。また、プログラムの意味論などの応用概念についても基礎的な事項を学習する。

[修得項目]

ラムダ計算，型，関数の抽象的概念，リスト処理

[知識項目]

Church-Rosser の定理，コンビネータ，遅延評価，高階プログラミング

[関連項目]

表示の意味論，関数型言語の実現，並列関数型言語

講義項目

1. 関数 (型) プログラミング概論 [1]

概論なのでキーワードは特にないが，関数とは何か？について中心的に話す。

2. ラムダ計算 [2]

ラムダ記法，簡約化，正規形，カーリー化，Church-Rosser の定理

3. コンビネータ理論 [2]

コンビネータ，不動点定理，再帰関数

4. ラムダ計算と計算可能性 [1]

計算実行可能性 (effectively computable)，チューリング機械

5. ラムダ計算と型 [1]

型，型付きラムダ計算，表示の意味論

6. リスト処理 [2]

リスト，高階演算，S 式

7. LISP 入門 [2]

パターンマッチ，S 式，条件判定，quote

8. LISP とラムダ計算 [1]

定数付きラムダ計算，遅延評価，値呼び，参照呼び，名前呼び

9. LISP による高階関数 [1]

メタ関数，マクロ，自己反映計算

10. 関数型言語の実現 [1]

ガベージコレクション，無限リスト，ストリーム，データフロー，並列処理

11. その他の関数型言語 [2]

APL，FP，Scheme，ML，Milanda

先修科目

データ構造とアルゴリズム，形式言語とオートマトン，コンパイラ

後修科目

プログラム意味論，ソフトウェア工学

関連科目

論理設計，オペレーティングシステム，データベース，数理論理学

実験及び演習

- 演習は必須である．特にラムダ計算，コンビネータでは専用の演習用環境が必要となる．
- 演習時間はラムダ計算，コンビネータ理論，リスト処理，LISP で各 1 時間ずつ

実施方法

- ラムダ計算と計算可能性，ラムダ計算と型が難しいようならば省略して，関数型言語の実現を 2 時間にする．

教科書/参考書

- Common Lisp 入門，萩谷 昌己，湯浅 太一 著，岩波書店，1996 (教科書)
Lisp の入門書である．1～7 章までを教える．
- 計算モデルの基礎理論，井田哲雄 著，岩波書店，1991 (教科書)
ラムダ計算，関数型計算モデルの入門部 1～4 章までを教える．
- 関数プログラミング，R. Bird，P. Wadler 著，武市 正人 訳，近代科学社 1991 (参考書)
関数プログラミングの様々な概念に触れているが，多少難しい．

G-8 オブジェクト指向プログラミング [1]

目的

大学院での講義ではあるが、基本的にオブジェクト指向プログラミングそのものが実用性指向なので、学習者が継承やポリモルフィズム、クラスライブラリ等を利用したプログラミングができるようにする。

概要

オブジェクト指向プログラミングについて、抽象データ型、メッセージング、継承、ポリモルフィズムといった基本的概念を実習を通じて学ぶ。また、実用的なオブジェクト指向プログラミングを行えるようにする。

[修得項目]

ソフトウェア設計、オブジェクト、クラス、メッセージ、継承、ポリモルフィズム

[知識項目]

ソフトウェア部品、インタフェースと実現の分離、フレームワーク(クラスライブラリ)、メタ(高階)プログラミング

[関連項目]

永続性、オブジェクト指向言語の実現、並列オブジェクト指向言語

講義項目

1. オブジェクト指向プログラミング概論 [2]

オブジェクト指向言語の歴史、オブジェクト、カプセル化、メッセージング

2. 抽象データ型とオブジェクト [1]

抽象データ型、カプセル化

3. 継承 [2]

プログラムの再利用、継承

4. ポリモルフィズム [1]

メッセージング、ポリモルフィズム

5. オブジェクト指向分析・設計 [2]

抽象データ型、型理論、表明、Booch 法、OMT 法

6. オブジェクト指向言語入門 [1]

演習で使用するオブジェクト指向言語の入門編

7. フレームワーク [1]

クラスライブラリ

8. プログラミング演習 [2]

課題のプログラミング

9. オブジェクト指向言語の実現 [2]

ガベージコレクション，メッセージ，ストリーム，並列処理，永続性

10. 様々なオブジェクト指向言語 [2]

Smalltalk，JAVA，C++，Eiffel，Scheme，Objective-C 等の言語について，その設計思想や実現手法の違いについて学ぶ

11. オブジェクト指向言語の進んだ概念 [1]

多重継承，並列・並行オブジェクト，オブジェクト指向 OS，モバイルオブジェクト

先修科目

データ構造とアルゴリズム，形式言語とオートマトン，コンパイラ

後修科目

データベース，ソフトウェア工学

関連科目

関数プログラミング，論理設計，オペレーティングシステム，プログラム意味論，数理論理学，集合論

実験及び演習

- 演習は必須である．特に初めの何時間かは特別なプログラミングをしなくとも即座に利用できるオブジェクト指向言語環境が必要である．
- 演習時間は演習と明記したものを除いて，継承，ポリモルフィズム，オブジェクト指向言語入門で各 1 時間ずつ

実施方法

- オブジェクト指向分析・設計を無くして，演習時間を増やしてもよい．

教科書/参考書

- オブジェクト指向プログラミング入門，T. A. Budd 著，羽部正義訳，アジソンウェスレイ・トッパン，1992 (教科書)
概念に関する入門だけでなく，様々なプログラミング言語にも触れているので，この講義には最適の教科書．
- オブジェクト指向のプログラミング改訂第 2 版，B. J. Cox/A. J. Novobilski 著，松本正雄 訳，アジソンウェスレイ・トッパン，1992 (参考書)
基本的な概念と，その応用に関してきっちりと押さえられている．Objective-C を用いた演習等に最適．

- オブジェクト指向入門, B. Meyer 著, 酒包 寛, 酒包 順子 訳, 二木 厚吉 監訳, アスキー出版局, 1990(参考書)
概念, 言語, 設計, クラスライブラリ, 実装とオブジェクト指向プログラミングの全てに関してまんべんなく触れている.
- オブジェクト指向コンピューティング, 所 真理雄, 松岡 聡, 垂水 浩幸 著, 岩波書店, 1993(教員用参考書)
オブジェクト指向プログラミングだけでなく, さらに進んだ概念等についてもまとめられており, プログラムの例も豊富. ただし, 入門書ではない.
- オブジェクト指向概論第2版, I. Graham 著, MISCO オブジェクト指向研究会訳, 木村俊夫監訳, アジソンウェスレイ・トッパン, 1996(教員用参考書)
オブジェクト指向設計に関する最新の動向についてまとめた本.

G-9 音声情報処理 [1]

目的

音声情報処理全般について概観する．音声の基本的な性質と特徴量について述べ，LPC 分析などの代表的な分析手法を説明する．音声符号化，音声合成，及び音声認識についてその概要を述べる．音声認識においては，HMM に基づく音素モデル，N-gram などの言語モデルについて説明する．また，言語モデルの情報理論的な評価法も述べる．

概要

音声情報処理の概要を学ぶ．音声の基本的な性質と特徴量，及びデジタル信号処理に基づく分析手法を修得する．音声符号化，音声合成，及び音声認識についてその概要を把握する．また，言語モデルの情報理論的な評価法も修得する．

[修得項目]

音声の基本的性質，線形分離等価モデル，周波数分析，ピッチ抽出，LPC 分析，音声符号化の概要，ベクトル量子化，音声合成の概要，音声認識の概要，音素モデルの構成法，隠れマルコフモデル (HMM)，トレリス計算，言語モデルのパープレキシティ，N-gram

[知識項目]

ケプストラム，AR モデル，LPC ケプストラム，波形編集，パラメータ合成，分析合成，規則合成，前向きアルゴリズム，Viterbi アルゴリズム，Baum-Welch アルゴリズム，離散分布 HMM，連続分布 HMM，クロスエントロピ，テストセットパープレキシティ，削除補間法，バックオフスムージング，有限状態オートマトン，文脈自由文法，One-Pass アルゴリズム，ビーム探索，A*探索，ワードスポッティング，音声理解・音声対話，話者認識

[関連項目]

スペクトル距離尺度，CELP，基本周波数モデル，声質変換，単語認識，連結学習，話者・環境適応，N-ベスト探索

講義項目

1. 音声の基本的性質 [2]

音素の種類，調音機構，線形分離等価モデル

2. 音声の特徴量 [2]

周波数分析，ケプストラム分析，ピッチ抽出

3. 音声の分析 [1]

LPC 分析，AR モデル，LPC ケプストラム，スペクトル距離尺度

4. 音声の符号化 [1]

ベクトル量子化，時間領域での符号化，周波数領域での符号化，CELP

5. 音声の合成 [1]

合成単位, 波形編集, パラメータ合成, 分析合成, 規則合成, 基本周波数モデル, 声質変換

6. 音声の認識 [1]

認識のタスク, 認識系の構成

7. 音素モデル [2]

隠れマルコフモデル (HMM), トレリス計算, 前向きアルゴリズム, Viterbi アルゴリズム, Baum-Welch アルゴリズム, 離散分布 HMM, 連続分布 HMM, 単語認識, 連結学習, 話者・環境適応

8. 言語モデル [2]

パープレキシティ, クロスエントロピー, テストセットパープレキシティ, N-gram, 削除補間法, バックオフスムージング, 有限状態オートマトン, 文脈自由文法

9. 探索アルゴリズム [1]

One-Pass アルゴリズム, ビーム探索, A*探索, N-ベスト探索, ワードスポッティング

10. 音声理解・音声対話 [1]

音声理解・音声対話システムの構成

11. 話者認識 [1]

話者識別, 話者照合, テキスト依存型, テキスト独立型

先修科目

信号処理, 情報代数と符号理論, 確率論と情報理論, 数理計画法, パターン認識, 形式言語とオートマトン, 人工知能

後修科目

なし

関連科目

デジタル通信, ヒューマン-コンピュータ・インタラクション, 画像情報処理, 画像情報処理特論, 自然言語処理, 集積回路工学

実施方法

- 特に後半は, 全部をカバーする必要はなく, トピックを選択すればよいと思われる。

教科書/参考書

- 音響・音声工学，古井貞熙 著，近代科学社，1992 (教科書)
ほとんどすべての項目を一通り網羅している．
- 音声，中田和男 著，コロナ社，1995 (参考書)
音声の分析について詳しく記述している．
- 確率モデルによる音声認識，中川聖一 著，電子情報通信学会・コロナ社，1988 (参考書)
音声の認識について詳しく記述している．
- 音声言語処理，北研二，中村哲，永田昌明 共著，森北出版，1996 (参考書)
言語モデルについて詳しく記述している．

備考 (コメント)

- 文法による言語モデルは「自然言語処理」との関連に留意する．統計的言語モデルは，情報理論の特論としての意義もある．

G-10 アルゴリズム論 [1]

目的

コンピュータ・サイエンスの研究をしていく上で効率のよいアルゴリズムは欠かせない。本科目では、各種例題を通して、よいアルゴリズムを開発するために必要な、アルゴリズムの設計法と解析法を修得させる。さらに、計算量理論の基礎を教え、問題の計算の難しさを解析する手法も学ばせる。

概要

効率のよいアルゴリズムを開発するための手法を、各種例題を通して修得する。また、アルゴリズムの効率の解析法や、さらには問題の難しさの解析法についても学ぶ。

[修得項目]

様々なアルゴリズムの設計手法(データ構造,乱数を利用したアルゴリズム,平均的に優れたアルゴリズム)の有効性,問題の難しさの測り方

[知識項目]

各種データ構造,動的計画法,乱数を利用したアルゴリズム,階層定理, NP 問題, NP 完全問題

[関連項目]

グリーディ・アルゴリズム, ヒューリスティクス各種, ならし効率, 平均効率

講義項目

1. データ構造の有効性 [3]

ヒープ, バランス木, ハッシュ

2. 効率のよいアルゴリズムを作るための手法 [4]

動的計画法, ならし効率のよいアルゴリズム, 乱数を利用したアルゴリズム, 平均的に効率のよいアルゴリズム

3. 計算の複雑さの理論への導入 [3]

アルゴリズムの効率解析, 問題の複雑さの定義, 階層定理

4. NP 問題 [3]

NP 問題の定義と例, 多項式時間還元, NP 完全性

先修科目

離散数学, 計算論, プログラミング言語論, データ構造とアルゴリズム

関連科目

情報セキュリティ

実施方法

- アルゴリズムの技法にはいろいろなものがあるので、それぞれの技法を習熟させても意味がない。それよりも、アルゴリズムには様々な考え方があること、そして、適切な手法（たとえば、適切なデータ構造）を用いることで効率が飛躍的に改良できること、などに気付かせることが重要である。

教科書/参考書

- Introduction to Algorithms, T. Cormen, C. Leiserson, R. Rivest 著, The MIT Press & McGraw-Hill, 1990 (参考書)
本科目の内容に、ほぼ即した構成になっている。他の話題も多い。ただし、計算の複雑さの理論に関する項目は少なく、たとえば、階層定理についての記述などが無い。
- 計算可能性・計算の複雑さ入門, 渡辺 治 著, 近代科学社, 1992 (参考書)
計算の複雑さの理論に関して、本科目の内容に、ほぼ即した構成になっている。
- Randomized Algorithms, R. Motwani and P. Raghavan, 1995 (教員用参考書)
乱数を利用したアルゴリズムや、アルゴリズムの平均効率の解析について、いろいろな例が紹介されている。
- The Design and Analysis of Algorithms, D. Kozen 著, Springer-Verlag, 1991 (教員用参考書)
データ構造に関するテーマを準備するときに参考になる。ただし、説明がかなり簡潔なため、教科書あるいは学生の参考書としては不適當。

付録IV：モデル履修コース（詳細）

情報機器工学 [A]

コンピュータ工学 [B]

ソフトウェア科学 [C]

ソフトウェア設計工学 [D]

知能情報学 [E]

マルチメディア情報処理 [F]

情報ネットワーク学 [G]

数理情報科学 [H]

人間情報科学 [I]

CS 外学科卒業生が CS 分野へ移行してくる際に必要な CS 最小セット [X]

CS 外専攻大学院生のための CS 最小セット [Y]

[A] 情報機器工学

目標

CSの基礎的な知識をベースとして、半導体集積回路技術を利用して次世代の各種情報機器の中核となる情報回路や情報システムを設計できるシステム LSI 技術者を育成する。CS をベースとした数学的なシステム記述法やその詳細化手法と集積回路による物理的な実現のそれぞれの特性を理解し、両者を結び付ける能力を獲得させる。

概要

21 世紀の情報産業や半導体産業を支えるシステムと集積回路の両者の知識を持った新しいタイプの技術者を育成することを目指す。半導体集積回路を主要なシステム実現手段とすることを前提とし、従来の CS の成果や技術の基礎的な部分を理解して新しいシステム LSI 設計に取り組める人材を育成する。CS に基づいた数学的な仕様の記述、詳細化、システム設計手法の基礎を身につけさせるとともに、集積回路による物理的な実現における基礎的な原理と種々の制約を理解させる。将来、物理的な実現手法と制約条件が変化した場合に、CS のベースとなるモデル化の変更が必要であることを理解させ、CS の諸技術を物理的に実現することができる能力を養う。

論理回路や半導体集積回路技術を中心としたハードウェア設計、アーキテクチャやコンパイラ、オペレーティングシステムなどの基本ソフトウェア技術を必須科目として教える。さらに、選択で、画像や音声の処理技術あるいは通信技術など応用分野として重要と思われる科目を 1 ないし 2 科目履修させておくことが望ましい。

本コースは、現在の日本の総合電機メーカーや家電・通信機器メーカーが、今後必要とすると考えられる人材の育成を行うことを狙う。このような技術分野は必ずしも整理されているわけではないので、教える側の努力や工夫も必要である。また、今後の技術展開によって必要とされる知識も大きく変化する可能性があるため、基礎的な学力と学生一人一人が新しいシステム設計手法を考えていくための基礎的な能力を身につけさせるようにする必要がある。このために、CS の成果や技術に関する学習においても、それぞれの技術の背景にある種々の制約条件などを理解させ、新しい条件のもとでの柔軟な発想の転換ができるように教育することが重要である。さらに、実現手段である集積回路技術に対してもブラックボックスとしてではなく自らコントロールできる設計対象であると認識させることが重要である。この点に関しては、東京大学に設置された大規模集積システム設計教育研究センター (VDEC) などの新しい仕組みを最大限に利用した新しい教育手法の開発が緊急の課題である。

卒業者の進路

システム LSI 設計者、コンピュータハードウェア設計者、情報機器設計者、設計支援技術者。

履修科目

- M-3 計算論
- U-2 形式言語とオートマトン
- U-5 プログラミング言語論
- U-1 論理回路
- U-3 データ構造とアルゴリズム
- U-4 コンピュータアーキテクチャ

U-6 論理設計

U-8 コンパイラ

U-7 オペレーティングシステム

U-15 集積回路工学

[B] コンピュータ工学

目標

情報基盤を支えるコンピュータシステムについて、それを設計・合成・開発・応用(適用)する工学的センスを身につけることを目指す。

概要

コンピュータシステムをブラックボックスとしてとらえるのではなく、それを構成する基本機能について学習する。コンピュータシステムを構成するハードウェア機構とソフトウェア機能の各原理について学ぶ。ハードウェアとソフトウェアのインタフェースであるコンピュータアーキテクチャを中心に、論理回路からヒューマン・コンピュータ・インタラクションまでの情報処理過程を構成する中核機能およびそれを支える基本原理について学習し、人間(ヒューマン)が活用できる情報基盤あるいはそれを支えるコンピュータシステムの設計能力さらにはその応用能力を身につけることを目指す。

卒業者の進路

コンピュータメーカ(特に、コンピュータシステムの設計部門)、情報基盤の設計者、情報処理産業一般の設計/開発部門、コンピュータシステムの応用部門。

履修科目

- [M-2] 離散数学
- [M-3] 計算論
- [U-2] 形式言語とオートマトン
- [U-1] 論理回路
- [U-4] コンピュータアーキテクチャ
- [U-3] データ構造とアルゴリズム
- [U-5] プログラミング言語論
- [U-8] コンパイラ
- [U-7] オペレーティングシステム
- [U-12] 情報ネットワーク
- [U-19] ヒューマン-コンピュータ・インタラクション

[C] ソフトウェア科学

目標

高性能なプログラムをいかに作り、使いやすい形で利用者に提供するか、コンピュータを応用した情報処理システムをいかに構築するかなどを多方面から分析し、そのようなソフトウェアシステムを作成できる人材を育てるコースである。また、これらのソフトウェアシステムに内在する理論的な問題を抽出し、抽象的にモデル化し、理論的に解決する素養も育てる。

概要

基本ソフトウェアそのものである「オペレーティングシステム」「コンパイラ」「データベース」の科目を中心として、それらの対象を数学的に扱うための基礎理論となる「形式言語とオートマトン」、基本ソフトウェアは効率よく動作することが重要であるので、そのための知識および原理を「データ構造とアルゴリズム」で学ぶ。

基本ソフトウェアが動作する計算機に関する知識として「コンピュータアーキテクチャ」、様々な処理を記述するための基本的な言語に関する知識として「プログラミング言語論」を学ぶ。基本ソフトウェアは多人数で作成するのでそのときの方法論などを「ソフトウェア工学」で、基本ソフトウェアの操作性に関する観点を「ヒューマン・コンピュータ・インタラクション」で学ぶ。

また、最近の基本ソフトウェアはネットワークを介して分散的に動作するので、ネットワークに関する知識を「情報ネットワーク」で学ぶ。

卒業者の進路

基本ソフトウェアを作成する計算機製造業・ソフトウェア製造業など。また、大学や研究所の研究者。

履修科目

- M-2 離散数学
- M-3 計算論
- M-6 数理論理学
- U-2 形式言語とオートマトン
- U-3 データ構造とアルゴリズム
- U-5 プログラミング言語論
- U-4 コンピュータアーキテクチャ
- U-7 オペレーティングシステム
- U-8 コンパイラ
- U-10 データベース
- U-12 情報ネットワーク
- U-13 ソフトウェア工学
- U-19 ヒューマン・コンピュータ・インタラクション

[D] ソフトウェア設計工学

目標

ソフトウェア開発とは、与えられた問題をコンピュータを利用して解決することであると考える、ここでは問題解決に関する抽象化能力ならびに具体的な解決法の実現能力を養うことを目標とする。問題対象の認識、分析、記述、解決法の発見、解決法の実現と評価に必要な基礎知識と多様な問題対象領域に適用できる応用能力とを身につけさせる。

概要

ソフトウェアシステムの開発および応用に必要な知識と能力を養うために、コンピュータの動作原理や構成、計算の基礎、問題のモデル化や定式化のための数学的基礎などの基礎的な知識や素養を身に付け、その上で、ソフトウェアシステムの開発に必要な専門知識を習得する。システム構築の具体例として、オペレーティングシステムやコンパイラなどのシステムプログラムの構成法や実現法を学ぶ。更に、ソフトウェア開発に関わる問題点を認識するためにソフトウェア工学の話題を学習する。

- 基礎：M-2, M-3, M-6, U-1, U-2
- システム構築のための道具立て：U-3, U-4, U-5, U-10
- システム構築の具体例：U-7, U-8
- ソフトウェア開発に関する議論：U-13

卒業者の進路

コンピュータシステムの開発および応用に関わる研究開発部門、ならびに、業務部門一般。業種は問わないが、システム開発法の確立や支援環境整備、特定の応用分野におけるシステム開発およびシステム運用管理の統括、特定の応用分野とシステム開発との橋渡し、応用業務分析およびモデル化、システム開発の上流工程におけるシステム分析/設計、システム開発管理などに従事する。例えば、研究者、プログラマー、システムエンジニア、システムアーキテクト、ドメインエンジニア、コンサルタント。などの職種が考えられる。

履修科目

- M-2 離散数学
- M-3 計算論
- M-6 数理論理学
- U-1 論理回路
- U-2 形式言語とオートマトン
- U-3 データ構造とアルゴリズム
- U-4 コンピュータアーキテクチャ
- U-5 プログラミング言語論
- U-7 オペレーティングシステム

U-8 コンパイラ

U-10 データベース

U-13 ソフトウェア工学

関連大学院科目

A-8 ソフトウェア工学特論

G-3 プログラム意味論

G-7 関数プログラミング

G-8 オブジェクト指向プログラミング

G-10 アルゴリズム論

その他

抽象的な思考に慣れるため、また、社会とのかかわりに関する問題意識を持つために、哲学書や思想書や歴史書を読むことを勧める。システム構築を行うために必要なコミュニケーション能力を養うために、国語（日本語）や外国語の知識と十分な表現能力を身に付けることが必要である。

[E] 知能情報学

目標

人工知能及び知能情報学の基礎を習得させ、人間・社会と接点を持つ広範な応用分野へと展開する力を身につけさせることを目的とする。

概要

数理的な最低限の基礎科目を習得した後、人工知能を中心にパターン認識などの周辺科目を学習する。具体的には、計算論、情報理論、形式言語理論を習得させると共に、人工知能、パターン認識、画像処理、自然言語処理などの人間との高度なインタフェースを形成するための基本科目を履修させる。なお、コンピュータ、ネットワークに関しては、実習によって利用方法を知るにとどめ講義科目から外しているため、実際にシステムを開発する段階で適宜学習する必要がある。

卒業者の進路

本コースを修得した学生は、コンピュータ、情報通信、電気機器、などの企業における研究開発部門、あるいは大学、研究機関においてこの分野の教育者・研究者として活躍することを想定している。

履修科目

- [M-3] 計算論
- [M-4] 確率論と情報理論
- [U-2] 形式言語とオートマトン
- [U-3] データ構造とアルゴリズム
- [U-10] データベース
- [U-11] 人工知能
- [U-17] 画像情報処理
- [U-18] パターン認識
- [U-19] ヒューマン - コンピュータ・インタラクション
- [U-21] 自然言語処理

[F] マルチメディア情報処理

目標

今後の情報技術を特徴づける大きな潮流に情報機器のネットワーク化とマルチメディア化がある。本コースは各種分野における情報機器のマルチメディア化に対処するとともに、その中核技術の開発を担う技術者、研究者の養成を目的としている。

概要

本コースの対象者としては、高校程度の数学と学部における情報系数学および基本的なプログラミング手法を習得している者を想定しているが、このような理科系科目以外に、心理学、芸術系科目等に精通していれば、将来さらに幅広い展開が可能となろう。本コースの履修では、既存の情報通信技術を基盤として、マルチメディア処理に関わる信号/画像処理、音声処理、グラフィックスの各技術を幅広く習得することを目指している。これは、将来の情報機器がマルチモーダルシステムへと進化し、真のマルチメディア時代が到来することを想定して、各種メディアの処理技術の基礎を学ぶためである。

卒業者の進路

コンピュータメーカ、電機メーカ等の情報機器および応用ソフトウェア研究開発部門、通信・放送・映画分野でのマルチメディアコンテンツの設計者。

履修科目

- [M-4] 確率論と情報理論
- [M-5] 数理計画法
- [U-3] データ構造とアルゴリズム
- [U-4] コンピュータアーキテクチャ
- [U-12] 情報ネットワーク
- [U-14] 数値計算
- [U-16] 信号処理
- [U-18] パターン認識
- [U-17] 画像情報処理
- [U-20] コンピュータグラフィックス
- [U-19] ヒューマン・コンピュータ・インタラクション

関連大学院科目

- [A-7] 信号処理特論
- [A-5] 画像情報処理特論
- [A-6] コンピュータグラフィックス特論
- [G-9] 音声情報処理

[G] 情報ネットワーク学

目標

ネットワーク関連産業界は常に即戦力を必要としている。しかし、情報ネットワークは成長を続けており、ある時点で最良とされたネットワーク技術が市場を支配し続けたことがないのは、その歴史が教える通りである。本コースでは、ネットワークの普遍的な原理をまず修得し、それに基づいた実現技術に関する知識を幅広く会得する。その結果、その時代の関連技術の成熟度に応じた情報ネットワークを構築できるような基礎体力を養うことを目標とする。

概要

本コースでは、情報ネットワークを設計し、構築していく上で基礎となるネットワークアーキテクチャの概念を理解し、ネットワークを介して情報がどのように伝送、処理されて相手に伝えられるのか、さらに情報通信機能を用いてどのようなサービスが実現されるのかを修得し、今後の情報ネットワークのありかたを展望するための基礎知識を会得する。そのためには、解析学、統計学などの数学に関する基礎知識は必須である。また、情報技術と通信技術の融合により、ネットワークを理解するにはコンピュータシステム/ソフトウェアに関する深い知識が要求されるようになっている。本コース取得者はそれらを踏まえて、本コースに望む必要がある。現状の、また、将来に渡るネットワーク技術の幅広い動向に関しては、大学院において別途コースを設ける。

卒業者の進路

情報ネットワークに関する知識や技術を有する人材は産業界において幅広く求められるようになっている。特にコンピュータネットワークの進展により、ネットワークに関する知識は本モデルコース修得者に限らず、CSを専門とする研究者/技術者にとって必須のものになっているが、特に、本コース修得者が中心となって活躍できる場としては、コンピュータメカのネットワーク関連開発部門やネットワーク関連機器メーカー、ネットワークサービス事業者などが挙げられる。また、大学院において本コースの上位コースを修得すれば、上記関連の研究部門が進路として加えられる。

履修科目

- M-1 情報代数と符号理論
- M-4 確率論と情報理論
- M-5 数理計画法
- U-1 論理回路
- U-3 データ構造とアルゴリズム
- U-16 信号処理
- U-9 デジタル通信
- U-4 コンピュータアーキテクチャ
- U-5 プログラミング言語論
- U-7 オペレーティングシステム

U-12 情報ネットワーク

U-17 画像情報処理

[H] 数理情報科学

目標

「情報」というキーワードは、ほとんどすべての産業・学問分野に関連している。しかし、種々雑多な情報の中から必要な情報を取り出し、価値のある情報へと加工する手法には、分野を問わず共通するものがある。特に、情報を数理的に解析する手法、それに情報を加工する CS の手法には、普遍的なものが多い。本コースは、情報を分析し加工するための数理科学的・CS 的手法の基礎を教育し、様々な分野で価値のある情報を作り出せる人材を養うことを目標とする。

概要

履修科目については、履修順序を考慮して列挙している。また、ここでは、基礎的な科目だけを列挙したが、選択的に信号処理、人工知能、情報ネットワークなど、応用分野科目の修得を勧めるとよい。

卒業者の進路

「情報」を扱う様々な職種。特に、情報を数学的に分析したり、情報を加工する技術を開発・提供する分野。大学院では、情報理論や離散数学などの情報数学の分野、あるいはオペレーションズリサーチ、理論 CS などの分野が適当だろう。

履修科目

M-2 離散数学

* 線形代数，解析学

U-5 プログラミング言語論

M-6 数理論理学

M-1 情報代数と符号理論

M-4 確率論と情報理論

* 関数解析（関数解析の基礎，ならびにフーリエ変換などの解析手法）

M-3 計算論

U-2 形式言語とオートマトン

U-3 データ構造とアルゴリズム

* 統計，ゲーム理論

M-5 数理計画法

U-8 コンパイラ

(注) * は J97 で挙げた科目の周辺領域に属する科目である。

[I] 人間情報科学

目標

今後のコンピュータ利用が益々盛んになることで、コンピュータと人間の相互作用（ヒューマン・コンピュータ・インタラクション，HCI）に関する基礎知識の重要性が増してくると思われる．本コースでは、コンピュータシステムの設計のうち、人間と関わる部分の開発に携わる技術者の養成を目的として、関連諸知識の習得を目標とする．

概要

本コースでは、コンピュータシステムの設計開発において人間と関わる部分を設計する技術者（エンジニア）が習得すべき関連科目を選択している．本コースを履修することにより、コンピュータシステムの利用者インタフェースに関連する技術項目の習得だけでなく、認知心理学，認知科学，社会学などのCS以外のHCIに関係する背景を持つ者と協力した設計開発を円滑に行うための背景知識の獲得を目指している．

卒業者の進路

コンピュータメーカ，電機メーカ，ソフトウェアメーカ等の情報機器および応用ソフトウェア研究開発部門．また，HCI 関連分野は広く，より広い知識の習得を目指す場合は，大学院へ進学することが望ましい．

履修科目

- [M-3] 計算論
- [M-4] 確率論と情報理論
- [U-3] データ構造とアルゴリズム
- [U-4] コンピュータアーキテクチャ
- [U-13] ソフトウェア工学
- [U-19] ヒューマン - コンピュータ・インタラクション
- [U-17] 画像情報処理
- [U-18] パターン認識
- [U-20] コンピュータグラフィックス
- [U-12] 情報ネットワーク
- [U-10] データベース
- [U-11] 人工知能
- [U-21] 自然言語処理

関連大学院科目

- A-4 データベース特論
- A-5 画像情報処理特論
- A-6 コンピュータグラフィックス特論
- A-8 ソフトウェア工学特論
- G-9 音声情報処理

[X] CS 外学科卒業生が CS 分野へ移行してくる際に必要な CS 最小セット

目標

本コースは，CS 外学科に所属しているあるいは卒業した学生が CS 分野へ移行（学部編入や大学院入学）してくる際に，その前提条件として，最小限必要な知識を修得することを目標とする．

概要

CS 分野は，コンピュータシステムそれ自体または，コンピュータシステムを利用する分野である．したがって，問題解決手法とそのアルゴリズム化を秩序だてて考え，コンピュータシステムでそのアルゴリズムを実行するための具体的手段であるプログラミング言語がどのようにしてコンピュータで実行されるのか，その原理・仕組みを理解・修得するのが，最低条件である．したがって，コンピュータシステムの各階層を構成している，プログラミング言語，コンパイラ，オペレーティングシステム，コンピュータアーキテクチャを履修しなければならない．また，これらに共通する科目であるデータ構造とアルゴリズムも履修しなければならない．本コースでは，必要最低限の履修科目のみを示すが，大学院入学の際には，これに加えて，自分が専攻したい CS 内の専門分野の知識もある程度修得しておくことが望ましい．

履修科目

- U-3 データ構造とアルゴリズム
- U-5 プログラミング言語論
- U-4 コンピュータアーキテクチャ
- U-8 コンパイラ
- U-7 オペレーティングシステム

[Y] CS 外専攻大学院生のための CS 最小セット

目標

副専攻を制度的に実施している大学は国内ではいまのところ少ないが、今後増えてくる可能性もあり、ここでひとつの案を提示することとしたい。実際、多様化が叫ばれており、現状においても、CS を副専攻として習得したいという学生のバックグラウンドも目標も多様であることはいうまでもない。そのことを承知の上で、あえてカリキュラム案を示すという要求に答えてみたものである。

概要

この案は主専攻が工学系、理科系である場合を想定した。日常的にコンピュータを使用しており、ある程度コンピュータの基礎知識は断片的に持っていることが多い。そういう学生には、学問的に系統だった内容を学習してもらう方が新鮮味もあり、自分のもっている断片的な知識の位置付けもできる。また、学部の講義であっても、大学院生の知的マチュリティにあった、知的興味を満足させるものの方が、効果が期待できる。また、この分野特有の思考法やアプローチが伝えられるであろう。オペレーティングシステムは実際に世話になっているものでもあり、かつ種々のプログラム技術や技法の集大成とも言うべきものであって、コンピュータアーキテクチャがハードウェア寄り、オペレーティングシステムがソフトウェア寄りで相互に補いあって、コンピュータシステムの全体像が把握できるであろう。

履修科目

- [L-2] プログラミング入門
- [U-2] 形式言語とオートマトン または、[M-3] 計算論
- [U-3] データ構造とアルゴリズム
- [U-4] コンピュータアーキテクチャ
- [U-7] オペレーティングシステム

- 10 単位 ~ 15 単位程度

検討

以上一つの案を示した。ただし、読者もこの案については、いろいろのご意見をおもちであろう。これを無理に勧めるということではない。多様なバックグラウンドと意識、目的をもつ学生に対して、本来行なうべきことは、画一的なカリキュラムを押し付けるのではなく、その学生個人に向けた個人ごとのメニューをアドバイザーが調製して示すことである。特に大学院の副専攻においては、きめ細かいアドバイスをすべきであろう。

実施方法については、学部で開講している科目を指定し、それを聴講させるというのが一つの案である。ただ、大学院で副専攻として CS を選択する学生のニーズはいくつかのタイプがある。いくつか思いつくタイプをあげてみよう。

1. 単に自分の主専攻でコンピュータを使ってデータ解析を行ったり、統計処理を行って、グラフを描くというような応用が多いので、それにもつぱら興味があるという学生。
2. CS という学問分野の特徴的な科目を学習し、自分の分野との境界領域を探求していくことを目指す学生。

3. コンピュータ関連の知識をもつことで、就職に有利になるという期待をもって選択した。そのことをアピールできるように、副専攻であってもできるだけ多くの知識やスキルを身につけたいと希望する学生。

ここに、あげた 1, 3 のタイプの場合は、いわゆる即戦力になるような期待を大きくもっている。そういう学生に上の科目群がそのまますんなりと受け入れられるかは多少疑問が残る。もちろん、変化の激しいこの分野で特定のソフトウェアやハードウェアについての知識は短寿命であり、真にこの分野のエッセンスを学ぶなら、基礎的なこれらの科目を学ぶのが良いということを十分に説明するにしても、説得力はやや弱いかもしれない。一つの案は、上記科目をそのまま取り上げるのではなく、むしろエッセンスを圧縮して時間数を減らし、各人がある程度希望のアドバンスドコースをとれるように計らった方が、効果的かもしれない。副専攻でなく、他分野からの移行学生に対するそのような講義の提案がある [1]。これは 7 つの科目を二つの科目に圧縮したものである。講義の単位数 20、実験 6 の計 26 単位を前期 5 単位、後期 5 単位に圧縮している。あいた時間で他の大学院レベルの専門科目もとれるように考慮している。もちろん、この方式は別の科目を新設するので教育負担が増えるとか、種々の問題があるものの教育効果という面でのメリットは大きいと報告している。

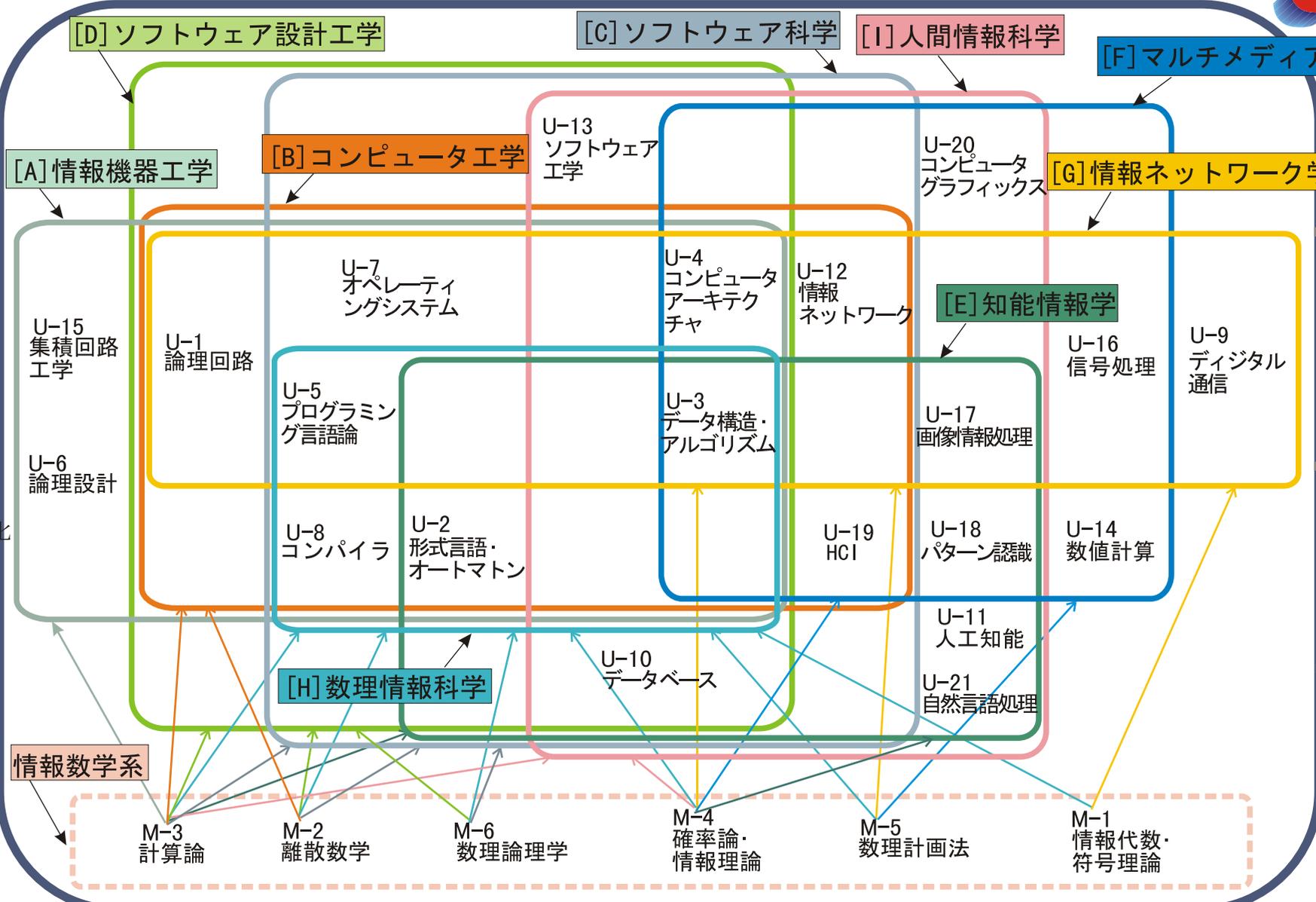
副専攻学生に対して、より効果的な科目群とその内容、教育方法は今後の検討課題という状況というべきであろう。

[1] J.D.Starkey, R.S.Babcook and A.S.DeFrance: An Accelerated Introductory Computer Science Course Sequence for Non-Traditional Master's Students, SIGCSE'97,pp.145-149. 1997.

コンピュータサイエンス (CS)

J97

応用
開発工学
↑
合成
↓
設計
↓
抽象化
モデル化
↓
解析
↓
理論
基礎科学



境界領域への進出

- 新しい学問領域**
 - ・計算代数学
 - ・計算物理
 - ・計算科学
 - ・ゲノム情報
 - ・発見科学
- メディア技術**
 - ・仮想現実感
 - ・拡張現実感
 - ・感性情報処理
 - ・ヒューマンインタフェース
- 社会システム**
 - ・電子商取引
 - ・電子図書館
 - ・情報倫理
 - ・知的所有権
 - ・CAI
- 脳と知能**
 - ・認知科学
 - ・脳科学
 - ・神経科学
 - ・複雑系
 - ・人工生命

