



コンピュータ科学領域 (J07-CS)

Computer Science Discipline (J07-CS)

J07

疋田輝雄 明治大学理工学部情報科学科

コンピュータ科学カリキュラム標準 J07-CS は、1997 年の J97 すなわち「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」の後継である。米国のコンピューティングカリキュラムコンピュータ科学 CC2001CS を形式および内容の面から参考にし、これに多数の変更と追加を加えたものである。カリキュラム標準を規定するにはまず、コンピュータ科学知識体系 CS-BOK-J 2007 を規定する。知識体系は専門エリア、ユニット、トピックの 3 レベルからなる。専門エリア (area) は 15 分野からなり、さらに、エリアを構成する各ユニットは、トピックと、ユニットの学習成果からなる。エリアやユニットの内容は、米国案をもとにして、日本の技術および大学教育の実状と最新技術の動向を踏まえて多数の改変を加えた。個々の授業科目 (course) はこの知識体系をもとにして、ユニットを組み合わせる。今回の CS カリキュラム標準の目標は、J97 の後継として、既存の理工系情報学科を想定することに加えて、国際的な整合性、日本の科学技術の特徴を活かすこと、および最新技術への考慮である。カリキュラム標準の提示形式の上でこれまでの J97 との違いは、米国版にならって、科目ではなく学問的な知識体系 (Body of Knowledge) を最初に与えることと、コアユニットとして必修の項目を導入したことである。成果主義教育としての各ユニットにおける学習成果が設定されている。さらに、24 の具体的な科目構成例が提示されている。

コンピュータ科学カリキュラム標準

コンピュータ科学カリキュラム標準 J07-CS

コンピュータ科学カリキュラム標準 J07-CS は、情報処理学会の 1997 年の J97 すなわち「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」²⁾の後継である。米国のコンピューティングカリキュラムコンピュータ科学 CC2001CS⁶⁾ を形式および内容の面から参考にし、これに多数の変更と追加を加えたものである。

カリキュラム標準を規定するにはまず、コンピュータ科学知識体系 CS-BOK-J 2007 を規定する。すなわち知識体系とは、学び方ないし教え方を捨象して、学生が学習すべき内容を独立して定めたものである。

知識体系は専門エリア、ユニット、トピックの 3 レベルからなる。専門エリア (area) は 15 分野からなり、さらに、エリアを構成するユニットは、トピックと、ユニットの学習成果からなる。エリアやユニットの内容は、米国案をもとにして、日本の技術および大学教育の実状と最新技術の動向を踏まえて多数の改変を加えた。個々の授業科目 (course) はこの知識体系をもとにして、ユニットを組み合わせる。

今回の CS カリキュラム標準の目標は、J97 の後継とし

て、既存の理系情報学科を想定することに加えて、国際的な整合性、日本の科学技術の特長を活かすこと、および最新技術への考慮である。カリキュラム標準の提示形式の上でこれまでの J97 との大きな違いは、米国版にならって、科目ではなく学問的な知識体系 (Body of Knowledge) を最初に与えることと、コアユニットとして必修の項目を新たに導入したことである。また昨夏に発行された中間報告³⁾への追加として、今回の報告案⁴⁾に加えるものは、成果主義教育としての各ユニットにおける学習成果の設定と、具体的な科目サンプルの提示である。

カリキュラム標準は、基準や規約と見なすべきものではなくて、モデルと見なすべきものである。同様に、ユニットから構成する科目は、例であって基準というものではない。なお、コンピュータ科学(サイエンス)という語についてであるが、知識体系の具体的な内容を見ても分かるが、英語での用語サイエンスは、日本語における用語の科学ないし理学よりも広い範囲を指すと考えてよい。

歴史

米国における情報科学技術分野での最初のカリキュラムモデルは、1968 年のカリキュラム 68 である⁵⁾。その後およそ 10 年ごとに(1968, 1978, 1991, 2001)、カリキュラムモデルが改定され提示されている。これらははじめは科目を

提示していたが、1991年版からは知識体系を示している。最新のものとしては、5つの分野においてそれぞれ提示され、コンピュータ科学分野では2001年に Computing Curricula 2001 Computer Science が示された。

情報処理学会は1990年に大学情報系学科におけるコンピュータ科学の専門教育コアカリキュラムを提言した。さらに1997年のJ97では、「コンピュータサイエンス」は、この分野の多様性から、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用いる。また、理論と実際、モデルとインプリメント、解析と合成など、いわば理と工の両面を重視するとしている。コア科目の指定はなく、各学科ごとに科目を選び出すものとする。講義科目としては、リテラシー科目として2科目、情報数学系科目として5科目、一般の(専門)科目として21科目を提示している。これら以外に大学院科目として18科目も示している。

さらに、理工系情報学科の多様性に対応するため、例示するモデル履修コースを9種あげ、コースごとに適した科目を示している。9種のコースとは、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、人間情報科学である。

コンピュータ科学の特徴と目標

コンピュータ科学 (Computer Science) は、情報処理とコンピュータに関する、基本的であるとされる諸領域 (area) を系統的に扱う、教育・研究分野である。CS のカリキュラムモデルは、米国および我が国において、1968年のACMカリキュラム68以来、提示されてきた。我が国の大学理工系情報学科はCS分野である学科が多いが、米国CS学科と同じというわけではなく、より多様である。

コンピュータ科学は、情報とコンピュータの理論的系統的な扱いを主とする。このことによりコンピュータ科学は、それ自身としての重要性とともに、情報の諸分野の基礎としての役割を持つ。またコンピュータ科学は広く他の諸科学工学、さらには文系諸科学への情報・コンピュータ応用における基礎としての役割を持つ。

なお米国の大学教育プログラム認証機関であるCAC (Computing Accreditation Commission) での記述を参考にすると、修得すべき知識・能力として、学習・教育目標には次の(1)、(2)を具体化したものが含まれていなければならない。

- (1) コンピュータを用いたシステムのモデル化および設計に、数学的な基礎、アルゴリズムの諸原理および情報科学の諸理論を応用する能力。
- (2) さまざまな複雑性を有するソフトウェアシステムの構築に、設計や開発の諸原理を応用する能力。

すなわち産業界において実際に情報システムに携わる設計者や実務担当者が基本的な素養として持っていなければならない知識と技術を表している。

J97 との比較

カリキュラム J97 との相違点は、科目ではなく知識体系を与えること、ユニットごとの学習成果の明示、そして、コアユニットとして必修の考え方を導入したことである。また J97 と異なり、今回の提言では、CS 以外の4分野と並んでカリキュラムの提示を行う。コンピュータ科学という名称により即した内容の提案であるといえる。

米国 CC2001CS との比較

知識体系 CS-BOK-J は、エリア数 15、ユニット総数 138 である。米国知識体系 CC2001CS では 14 エリアであり、専門エリアとしては、マルチメディア表現 MR というエリアを新設している。MR では、コアユニットとして情報のデジタル表現と文字コード、およびその他のマルチメディアに関するユニットがある。

知識体系 CS-BOK-J2007

方針と特徴

知識体系とは、教育を目的として、教育内容を学問的に体系化したものである。知識体系 CS-BOK-J の作成の基本的な方針は次のとおりである。

- a) J97 の後継としての、多くの理工系情報学科を想定したカリキュラム標準
- b) 国際共通性、特に米国カリキュラムモデル CC2001CS との整合性
- c) 日本の科学技術の特長と独自性を活かす
- d) 最新技術動向への考慮

a) の J97 の後継としての J07-CS 特に CS-BOK-J については、今回の作成作業において最も留意した点である。これまでの J97 からのスムーズな移行と、日本の情報科学科の実質を考慮した。一方、カリキュラムモデルは、世のさまざまな工学的な基準と同様に、これまで以上に国際的な整合性が要求されていることも重視する。c) と d) については説明を要しないであろう。

コンピュータ科学知識体系 CS-BOK-J 2007

15の専門エリアからなる。各エリアはユニットからなり、ユニットの総数は138である。ユニットでは、その内容としてトピックが列挙され、そしてユニットとしての学習成果がいくつか指定される。図-1に、コンピュータ科学知識体系 CS-BOK-J のエリアとユニットの一覧を示す。

コンピュータ科学知識体系	
<p>DS 離散構造 (41)</p> <ul style="list-style-type: none"> ● * DS1 関数, 関係, 集合 (6) ● * DS2 論理 (6) ● * DS3 グラフ (4) ● * DS4 証明技法 (8) ● * DS5 数え上げと離散確率の基礎 (7) ● * DS6 オートマトンと正規表現 (6) ● * DS7 計算論概論 (4) ○ DS8 計算論 	<ul style="list-style-type: none"> ○ HC3 人間中心のソフトウェア評価 ○ HC4 人間中心のソフトウェア開発 ○ HC5 グラフィカル・ユーザインタフェースの設計 ○ HC6 グラフィカル・ユーザインタフェースのプログラミング ○ HC7 マルチメディアシステムのHCI的側面 ○ HC8 協同作業とコミュニケーションのHCI的側面
<p>PF プログラミングの基礎 (38)</p> <ul style="list-style-type: none"> ● * PF1 プログラミングの基本的構成要素 (9) ● * PF2 アルゴリズムと問題解決 (6) ● * PF3 基本データ構造 (14) ● * PF4 再帰 (5) ● * PF5 イベント駆動プログラミング (4) 	<p>MR マルチメディア表現 (3)</p> <ul style="list-style-type: none"> ● * MR1 情報のデジタル表現 (2) ● * MR2 文字コード (1) ○ MR3 標準化, 量子化, 圧縮の原理とアルゴリズム ○ MR4 マルチメディア機器 ○ MR5 オーサリング
<p>AL アルゴリズム (20)</p> <ul style="list-style-type: none"> ● * AL1 アルゴリズムの解析の基礎 (4) ● * AL2 アルゴリズム設計手法 (8) ● * AL3 アルゴリズム設計例 (8) ○ AL4 アルゴリズムの高度な解析 ○ AL5 高度なアルゴリズムの設計 ○ AL6 計算量クラス P と NP ○ AL7 暗号アルゴリズム ○ AL8 幾何アルゴリズム ○ AL9 データ分析アルゴリズム ○ AL10 並列・分散アルゴリズム 	<p>GV グラフィックスとビジュアル・コンピューティング (3)</p> <ul style="list-style-type: none"> ● * GV1 グラフィックスにおける基礎技術 (2) ● * GV2 グラフィック・システム (1) ○ GV3 2次元画像の生成と加工 ○ GV4 モデリング ○ GV5 レンダリング ○ GV6 コンピュータ・アニメーション ○ GV7 視覚化 ○ GV8 仮想現実 (VR) ○ GV9 コンピュータ・ビジョン
<p>AR アーキテクチャと構成 (32)</p> <ul style="list-style-type: none"> ● * AR1 論理回路と論理システム (6) ● * AR2 データのマシンレベルでの表現 (2) ● * AR3 アセンブリレベルのマシン構成 (7) ● * AR4 メモリシステムの構成とアーキテクチャ (5) ● * AR5 インタフェースと通信 (3) ● * AR6 機能的構成 (7) ● * AR7 並列処理と様々なアーキテクチャ (2) ○ AR8 性能の向上 ○ AR9 ネットワークと分散システムのためのアーキテクチャ 	<p>IS インテリジェントシステム (5)</p> <ul style="list-style-type: none"> ● * IS1 インテリジェントシステムの基本的問題 (3) ● * IS2 探索および制約充足 (2) ○ IS3 知識表現および推論 ○ IS4 高度な探索 ○ IS5 高度な知識表現と推論 ○ IS6 エージェント ○ IS7 自然言語処理 ○ IS8 機械学習とニューラルネット ○ IS9 AI プランニングシステム ○ IS10 ロボット工学
<p>OS オペレーティングシステム (17)</p> <ul style="list-style-type: none"> ● * OS1 オペレーティングシステムの概要 (1) ● * OS2 利用者から見たオペレーティングシステム (1) ● * OS3 オペレーティングシステムの原理 (1) ● * OS4 プロセスの構造とスケジューリング (3) ● * OS5 並行性 (4) ● * OS6 メモリ管理 (4) ○ OS7 入出力デバイス管理と入出力 ● * OS8 ファイルシステム (2) ● * OS9 認証とアクセス制御 (1) ○ OS10 セキュリティと高信頼性 ○ OS11 リアルタイムシステムと組み込みシステム ○ OS12 並列・分散処理のためのオペレーティングシステムの機能 ○ OS13 オペレーティングシステム構成法 ○ OS14 システム性能評価 	<p>IM 情報管理 (14)</p> <ul style="list-style-type: none"> ● * IM1 情報モデルとシステム (2) ● * IM2 データベースシステム (2) ● * IM3 データモデリング (4) ● * IM4 関係データベース (3) ● * IM5 データベース問い合わせ言語 (3) ○ IM6 関係データベース設計とデータ操作 ○ IM7 トランザクション処理 ○ IM8 分散データベース ○ IM9 データベースの物理設計 ○ IM10 データマイニング ○ IM11 情報格納と情報検索 ○ IM12 ハイパーテキストとハイパーメディア ○ IM13 マルチメディアデータベース
<p>NC ネットワークコンピューティング (14)</p> <ul style="list-style-type: none"> ● * NC1 ネットワークコンピューティング入門 (2) ● * NC2 通信とネットワーク接続 (7) ● * NC3 ネットワークセキュリティ (2) ● * NC4 クライアントサーバコンピューティングの例としてのウェブ (3) ○ NC5 分散アプリケーションの構築 ○ NC6 ネットワーク管理 ○ NC7 ワイヤレスおよびモバイルコンピューティング ○ NC8 マルチメディア情報の配信システム 	<p>SP 社会的視点と情報倫理 (11)</p> <ul style="list-style-type: none"> ● * SP1 コンピュータの歴史 (1) ● * SP2 社会におけるコンピュータ (2) ○ SP3 倫理・価値判断の方法 ● * SP4 専門家としての倫理的責任 (3) ○ SP5 コンピュータ・ベース・システムのリスクと脆弱性 ● * SP6 知的財産権 (3) ● * SP7 プライバシーと市民的自由 (2) ○ SP8 コンピュータ犯罪 ○ SP9 コンピュータにおける経済問題 ○ SP10 哲学的枠組み
<p>PL プログラミング言語 (17)</p> <ul style="list-style-type: none"> ● * PL1 プログラミング言語の概要 (2) ● * PL2 仮想計算機 (1) ● * PL3 言語翻訳入門 (2) ● * PL4 宣言と型 (3) ● * PL5 抽象化メカニズム (3) ● * PL6 オブジェクト指向言語 (6) ○ PL7 関数型言語 ○ PL8 論理型言語 ○ PL9 スクリプト言語 ○ PL10 言語翻訳システム ○ PL11 型システム ○ PL12 プログラミング言語の意味論 ○ PL13 プログラミング言語の設計 	<p>SE ソフトウェア工学 (32)</p> <ul style="list-style-type: none"> ● * SE1 ソフトウェア設計 (8) ● * SE2 APIの使用 (5) ● * SE3 ソフトウェアツールおよび環境 (3) ● * SE4 ソフトウェアプロセス (2) ● * SE5 ソフトウェア要求および仕様 (5) ● * SE6 ソフトウェア妥当性検査 (3) ● * SE7 ソフトウェアの進化 (3) ● * SE8 ソフトウェアプロジェクト管理 (3) ○ SE9 コンポーネントベース開発 ○ SE10 形式手法 ○ SE11 ソフトウェアの信頼性 ○ SE12 専用システムの開発
<p>HC ヒューマンコンピュータインタラクション (8)</p> <ul style="list-style-type: none"> ● * HC1 ヒューマンコンピュータインタラクションの基礎 (6) ● * HC2 簡単なグラフィカル・ユーザインタフェースの構築 (2) 	<p>CN 計算科学と数値計算 (0)</p> <ul style="list-style-type: none"> ○ CN1 数値解析 ○ CN2 オペレーションズリサーチ ○ CN3 モデリングとシミュレーション ○ CN4 ハイパフォーマンス・コンピューティング <p>※エリア, ユニットとも, 行末の括弧内はコアの最低履修時間である。</p>

図-1 コンピュータ科学知識体系 CS-BOK-J のエリアとユニット一覧

NC4 クライアントサーバコンピューティングの例としてのウェブ

[コア最低履修時間 3 時間]

トピックス

クライアントサーバ関係の特徴
ウェブ技術
HTML と URI
ウェブプロトコル
サーバ側のプログラム
コモン・ゲートウェイ・インタフェース (CGI) プログラム
クライアントサイドスクリプト
サーバとクライアントの協調
アプレットの概念
ウェブサーバの特性
パーミッションの扱い
ファイル管理
一般的なサーバアーキテクチャの能力
ウェブサイト作成およびウェブ管理のためのサポートツール
インターネット情報サーバの開発例
情報やアプリケーションの公開例

学習成果

1. 複数のアプリケーションプログラムについて、クライアントとサーバの役割を説明できる。
2. さまざまなクライアントサーバ連携を効率的に実現するためツール群を選択できる。
3. 簡単な対話型ウェブベースアプリケーション (たとえば、クライアントから情報を集め、それをサーバ上のファイルに格納するウェブページ) を設計し、実装できる。

図-2 NC4：クライアントサーバコンピューティングの例としてのウェブ

コンピュータ科学の知識体系としてのこのような再構成は、学問的なものであり、同時に教育的なものでもある。つまり、現時点でのコンピュータ科学としての科学・工学的な構造を示すものであるが、必ずしも学問的な厳密性を追及したものではなく、教える順序などの、教育上の効率や便宜も考慮する。たとえば教育上で、入門としての役割だけを持つユニットがいくつかある。

ユニットのうちのいくつかはコアユニットである。これは図-1 では時間数の指定のあるユニットである。コアユニットは必修を表し、コンピュータ科学を履修するすべての学生がこれらのユニットは履修することを想定していることを示す。したがって、ユニットから授業科目を構成する際に、すべてのコアユニットは一式のカリキュラムのどこかの科目に必ず含まれていることが要請される。

なお、知識体系は専門分野についてのものであり、学部教育はこれらだけで済むものではない。BOK 以外の一般教育として、次の 5 つがある。

- 1) 数学(解析, 線形代数, 確率統計, 論理等)
- 2) 科学の能力
- 3) 実際分野への応用力
- 4) コミュニケーション能力(語学等)

MR2 文字コード

[コア最低履修時間 1 時間]

トピックス

文字の字形と符号化および文字コード
フォントとの関連付け
文字コードの国際規格

学習成果

1. 1 バイトで表現されたコードについて各国やベンダでの表現の違いを説明できる。
2. 日本語の複数の文字コードの違いを説明できる。
3. 国際的な文字コードの規格について説明できる。
4. 字形と文字コードとの対応付け、および文字コードの符号化の違いについて説明できる。

図-3 MR2：文字コード

SE5 ソフトウェア要求および仕様

[コア最低履修時間 5 時間]

トピックス

ステークホルダ分析と要求獲得
要求分析モデル化技法
機能要求および非機能要求
プロトタイプング
形式仕様技法の基礎的な概念

学習成果

1. ステークホルダ分析、および要求獲得の目的を説明できる。
2. ステークホルダ分析、要求獲得、および獲得した要求の分析を行うために、プロトタイプングをはじめとした各種手法、および形式的ではないモデル化手法を適用し、中規模のソフトウェアを開発するための要求仕様書を作成できる。
3. ソフトウェアの保守が困難になる理由を、要求仕様書の管理という点から説明できる。
4. 文書の品質を決定するために、過去の成功事例を適用してソフトウェア要求仕様書のレビューを行うことができる。
5. 広く使用されている形式仕様言語で書かれたソフトウェア要求仕様書を自然言語に変換できる。

図-4 SE5：ソフトウェア要求および仕様

5) チームとしての協調力

たとえば数学では、離散数学や論理の一部などは専門知識として知識体系内に規定するが、これら以外は知識体系には規定しない。

ユニットの内容および記述の具体的な例として、ネットワークコンピューティング、マルチメディア表現、ソフトウェア工学の 3 つの専門エリアから 1 つずつを紹介する。

NC4：「クライアントサーバコンピューティングの例としてのウェブ」

MR2：「文字コード」

SE5：「ソフトウェア要求および仕様」

の 3 つのユニットであり、いずれもコアユニットである (図-2, 3, 4)。

		ユニット数	コアユニット数	コア時間	コア時間(米国)
DS	離散構造	8	7	41	(43)
PF	プログラミングの基礎	5	5	38	(38)
AL	アルゴリズム	10	3	20	(31)
AR	アーキテクチャと構成	9	7	32	(36)
OS	オペレーティングシステム	14	8	17	(18)
NC	ネットワークコンピューティング	8	4	14	(15)
PL	プログラミング言語	13	6	17	(21)
HC	ヒューマンコンピュータインタラクション	8	2	8	(8)
MR	マルチメディア表現	5	2	3	—
GV	グラフィックスとビジュアル・コンピューティング	9	2	3	(3)
IS	インテリジェントシステム	10	2	5	(10)
IM	情報管理	13	5	14	(10)
SP	社会的視点と情報倫理	10	5	11	(16)
SE	ソフトウェア工学	12	8	32	(31)
CN	計算科学と数値計算	4	0	0	(0)
計		138	66	255	(280)

図-5 知識体系 CS-BOK-Jの各エリアのユニット数とコア時間

図-5に、知識体系CS-BOK-Jの15の専門エリアの、ユニット数、コアユニット数、コア講義時間、CC2001CSでのコア講義時間の一覧を示す。このコア講義時間とは、ユニットを講義として学習する場合での最低履修時間数を示す。

知識体系の内容およびコアユニットの指定については、さらに検討すべき点が多いと思われる。特にセキュリティに関してが今後の課題であろう。

学習成果

学習成果は、そのユニットの学習を終えた学生が持っている能力を示す。そのユニットを教える側の内容を示すのではないことが重要である。しかし学習成果は、結果としては教える側の規定にもなる。各ユニットにおける学習成果の設定は、特にコアユニット

の場合に、必修の内容を具体的に示すという意味で重要である。

米国との比較

知識体系CS-BOK-Jは、エリア数15、ユニット総数138である。米国知識体系CC2001CSでは14エリアであり、それに比べて、マルチメディア表現MRというエリアを新設している。そこでは、コアユニットとして情報のデジタル表現と文字コード、およびその他のマルチメディアに関するユニットがある。

コアユニットの講義としての総時間数は、米国CC2001CSのBOKでは280時間であり、CS-BOK-Jでは255時間である。これは日本の大学理工系における実状、特に卒業研究が時間的に大きい割合を占めていることに対応するものである。昨年3月のCS-BOK-J中間報告ではコア総時間数は240時間としていたが、その後のコア時間数の増加は、主にソフトウェア工学分野のコアユニットの増加による。

CC2001CSでは学習目的であり学生側の得た能力が教員側の内容かが曖昧であったが、CS-BOK-Jでは明確化した。

授業科目例の構成

方針

このカリキュラム標準では、個々の授業科目は、その大学および学科の目標、特徴や状況に応じて、ユニットをいくつか組み合わせて構成することを想定している。



この報告では、科目の構成の具体的な構成例として、24の科目を示す。これらはいくまで構成例であって、何らかの基準となるものではない。

15の各専門エリアに対応して、そのエリアでのコアユニットをすべてカバーするような、21の科目例を示す。ここでは、日本での多い例に応じて、90分15週の講義科目(2単位)として構成している。特に6つのエリアでは、2つの科目(4単位)となっていて、6つのエリアとは、離散構造、プログラミングの基礎、アルゴリズム、アーキテクチャと構成、ネットワークコンピューティング、ソフトウェア工学である。

またこれらとは別に、複数のエリアにまたがるような科目の例として、次の3つの科目の構成例を示している。これらは、入門的初等的な科目(B1, B2), または上級応用的な科目の例である(B3)。

B1: コンピュータ科学入門

B2: メディア・インタラクション

B3: データマイニング

専門科目での必修科目の単位数は、コアユニットのカバーの仕方や実現法(講義, 演習, 実習等)によるが、およそ40単位以上である。

ここでは講義, 演習, 実習, プロジェクト, 卒業研究などの形で授業を実現するかは、ユニットの内容や学習成果によって選ぶことになるが、この報告では講義の形で提示する。初歩のプログラミングなどは、講義に付属する演習や単独の演習や実習を行うことが望ましいのは当然である。

大学の学科がコアユニットをすべてカバーするようなカリキュラムを用意することは、学生への教育の最低限の保証を意味する。それ以上の、平均的レベルの教育、およびエキスパートレベルの教育のための科目は、この報告では十分には用意していないが、コアでないユニットをも適宜組み合わせることで実現される。

科目の表示

各科目は次の5項目によって構成されている。先修ユニット, 講義項目, 講義計画例(15回), カバーするコアユニット, そして教科書・参考書である。

図-6に科目例として、「コンピュータネットワーク」の記述を示す。この科目はエリア「ネットワークコンピューティング」のコアユニットをすべてカバーしている。先修ユニットはこのユニットを履修するためにあらかじめ必要なユニットである。ただしこれは履修者の便宜のための項目であり、これを厳格に捉えすぎてはならない。講義項目はユニットのトピックにあたるものであるが、履修者の便宜のためのものである。講義計画例は、15

回の授業のおよその配分を示す。カバーするコアユニットでは、場合によってはユニットの一部をカバーするときもある。教科書・参考書には、その講義の標準的教科書、あるいは入門書や参考書などさまざまな場合がある。履修者ないし講義をするための参考のためのものであり、標準であると指定しているわけではない。

カリキュラムの編成

図-7の学年配置表には、一般教育が含まれていないと同じく、応用科目、高度の科目や卒業研究がほとんど含まれないことを注意する。

コンピュータ科学教育委員会

2006～2007年度の委員会開催

2006年10月20日, 2007年1月12日, 12月7日, 12月25, 26日, 2008年1月26日, 2月27日。

委員リスト

委員長 疋田輝雄(明治大学)

幹事 石畑清(明治大学)

委員 板野肯三(筑波大学) 大岩元(慶應義塾大学) 角田博保(電気通信大学) 清水謙多郎(東京大学) 玉井哲雄(東京大学) 長崎等(共栄大学) 中里秀則(早稲田大学) 中谷多哉子(筑波大学) 野中誠(東洋大学) 三浦孝夫(法政大学) 箕原辰夫(千葉商科大学) 和田耕一(筑波大学) 渡辺治(東京工業大学)

参考文献

- 1) 情報処理学会: 大学等における情報処理教育のための調査研究報告書 (Mar. 1991).
- 2) 情報処理学会: 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 (Sep. 1999).
- 3) 情報処理学会情報処理教育委員会 J07 プロジェクト連絡委員会: 情報専門学科におけるカリキュラム標準 J07 (中間報告), 2007-07-31.
- 4) 情報処理学会コンピュータ科学教育委員会: コンピュータ科学知識体系 CS-BOK-J 2007, 2008-03-13. コンピュータ科学知識体系 CS-BOK-J 2007 カリキュラム例, 2008-03-13.
- 5) ACM Curriculum Committee on Computer Science: Curriculum '68: Recommendations for the Undergraduate Program in Computer Science, Comm. ACM, 11(3) pp.151-197 (1968).
- 6) The Joint Task Force on Computing Curricula: Computing Curricula 2001: Computer Science, Final Report, Dec. 15, 2001, IEEE Computer Society, ACM.

(平成20年6月2日受付)

疋田輝雄(正会員)

hikita@cs.meiji.ac.jp

1989年から明治大学理工学部情報科学科教授。計算理論、ネットワークコンピューティング等。2003年から本会コンピュータ科学教育委員会委員長。著書「コンパイラの理論と実現」(共立出版)他。

A10 コンピュータネットワーク

このコースは、コンピュータネットワークの構造と機能構成、セキュリティにかかわるアルゴリズム、およびコンピュータネットワークを使ったアプリケーションの典型的な例としてのウェブ技術など、コンピュータネットワーク技術について説明する。

先修ユニット

- PL1 プログラミング言語の概要
- MR2 文字コード

講義項目

- インターネット化とインターネットの背景と歴史
- ネットワークアーキテクチャ
- ネットワークコンピューティング分野の個別テーマの概要
- ネットワーク標準および標準化団体
- ISO 7 層参照モデルの一般論および TCP/IP におけるその具体例
- 回線交換とパケット交換, ストリームとデータグラム
- 物理層ネットワーク接続の概念, データリンク層の概念
- ネットワーク間接続とルーティング, トランスポート層サービス
- 暗号の基礎, 秘密鍵アルゴリズム, 公開鍵アルゴリズム
- 認証プロトコル, デジタル署名
- クライアントサーバ関係の特徴
- ウェブ技術, ウェブサーバの特性, ウェブサイト作成およびウェブ管理のためのサポートツール
- インターネット情報サーバの開発例, 情報やアプリケーションの公開例
- ネットワーク管理上の課題の概要
- パスワードおよびアクセス制御機構の使用
- ドメインネームとネームサービス
- インターネットサービスプロバイダ (ISP) にかかわる管理上の課題
- セキュリティとファイアウォール
- サービス品質の問題
- ワイヤレス標準の歴史, 発展, 互換性についての概観
- ワイヤレスおよびモバイルコンピューティングに固有な問題

講義計画例

1. ネットワークコンピューティング入門(ネットワーク化とインターネットの背景と歴史, ネットワークアーキテクチャ)
2. 通信とネットワーク接続(ネットワークコンピューティング分野の個別テーマの概要, ネットワーク標準および標準化団体)
3. 通信とネットワーク接続 (ISO 7 層参照モデルの一般論および TCP/IP におけるその具体例, 回線交換とパケット交換, ストリームとデータグラム)
4. 通信とネットワーク接続(物理層ネットワーク接続の概念, データリンク層の概念)
5. 通信とネットワーク接続(ネットワーク間接続とルーティング)
6. 通信とネットワーク接続(トランスポート層サービス)
7. ネットワークセキュリティ (暗号の基礎, 秘密鍵アルゴリズム, 公開鍵アルゴリズム)
8. ネットワークセキュリティ (認証プロトコル, デジタル署名, クライアントサーバ関係の特徴)
9. クライアントサーバ関係の特徴, ウェブ技術
10. ウェブサーバとウェブサイトの作成/管理
11. ネットワーク管理上の課題の概要, ドメインネームとネームサービス
12. パスワードおよびアクセス制御機構の使用, セキュリティとファイアウォール
13. インターネットサービスプロバイダ (ISP) にかかわる管理上の課題, サービス品質の問題
14. ワイヤレス標準の歴史, 発展, 互換性についての概観
15. ワイヤレスおよびモバイルコンピューティングに固有な問題

カバーするコアユニット

- NC1 ネットワークコンピューティング入門
- NC2 通信とネットワーク接続
- NC3 ネットワークセキュリティ
- NC4 クライアントサーバコンピューティングの例としてのウェブ

教科書・参考書

- コンピュータネットワーク (第 4 版), A. S. タネンバウム, 日経 BP 社.
- Data and Computer Communications (8th edition), W. Stallings, Prentice Hall.

講義科目名	BoK 専門エリア	1年 前期	1年 後期	2年 前期	2年 後期	3年 前期	3年 後期	4年 前期	4年 後期
A1 離散構造 (1)	DS 離散構造			○					
A2 離散構造 (2)	DS 離散構造				○				
A3 基礎プログラミング (1)	PF プログラミングの基礎	○							
A4 基礎プログラミング (2)	PF プログラミングの基礎		○						
A5 アルゴリズム (1)	AL アルゴリズム			○					
A6 アルゴリズム (2)	AL アルゴリズム				○				
A7 コンピュータシステム序論	AR アーキテクチャと構成		○						
A8 コンピュータアーキテクチャ	AR アーキテクチャと構成						○		
A9 オペレーティングシステム	OS オペレーティングシステム				○				
A10 コンピュータネットワーク	NC ネットワークコンピューティング					○			
A11 ウェブアプリケーション	NC ネットワークコンピューティング							○	
A12 プログラミング言語	PL プログラミング言語					○			
A13 ヒューマンコンピュータインタラクション	HC ヒューマンコンピュータインタラクション				○				
A14 マルチメディア表現論	MR マルチメディア表現					○			
A15 コンピュータグラフィックス	GV グラフィックスとビジュアル・コンピューティング							○	
A16 人工知能(インテリジェントシステム)	IS インテリジェントシステム							○	
A17 情報管理(データベース)	IM 情報管理					○			
A18 社会における情報技術	SP 社会的視点と情報倫理							○	
A19 ソフトウェア工学 (1)	SE ソフトウェア工学					○			
A20 ソフトウェア工学 (2)	SE ソフトウェア工学							○	
A21 数値計算	CN 計算科学と数値計算							○	
B1 コンピュータ科学入門		○							
B2 メディア・インタラクション			○						
B3 データマイニング									○

図-7 学年配置例

