

〔技術者教育〕 情報・通信分野に対するパブリックコメントへの回答

まず、次ページ以降に示す「前書き、コアカリキュラムについての考え方、情報・通信分野が想定している育成人材像、および今回のコアカリキュラム案作成のねらい」などを報告書の最初に I, II として追加して説明を加える。今回のパブリックコメントから、本報告書の狙い、育成する人材像、記載の形式が大学関係者に十分には理解されていないと判断できる。その対応策として、前書き等でこれらのことを提示して、よりの確な理解を得たい、ということがその意図である。そのあと III で、各コメントに対する回答を記載する。

(※ご意見をいただいた方についてはイニシャルと意見の投稿日時のみ記載)

I. まえがき

本報告は、技術者教育における情報・通信分野に焦点を当て、学士課程教育における教育カリキュラムのうち、コアカリキュラムと要望カリキュラムを提示する。すなわち、学士課程教育において身に付けるべき知識や理解、およびこれらを具体的事例に適用する能力（ここでは、運用力と呼ぶことにする）について、その範囲とレベルを、項目や到達目標の形で、学修に当たっての配慮事項とともに提示する。コアカリキュラムは、必ず履修すべき分野、項目からなり、必須の分野・項目である。要望カリキュラムは、できれば履修させたい分野・項目からなる。コアカリキュラムに含ませたいが、コアカリキュラムの内容が過多になることを避ける意味もあって、要望カリキュラムに含ませているものが多数ある。各々を、到達目標及び学修に当たっての配慮事項とともに記載している。

II. 本報告の狙いと意義、および記載の形式について

【コアカリキュラムの作成方針】

コアカリキュラムは、どの大学の学士課程教育においても必ず含むべき内容であり、ここでの多様性や曖昧性は極力避けるべきである。むしろ詳細に分野、項目、到達目標を記述しておくことが望ましい。要望カリキュラムは選択の可能性を持っている。各大学の教育方針、スタッフ、その他の要因を考慮して選択することになる。また、コアカリキュラム、要望カリキュラム以外にも選択肢は残っている。この辺りの選択肢に各大学の独自性、多様性、といった教育機関としての自主性は発揮できる。

学士課程での情報・通信分野の教育に求められる教育内容（分野や項目）の記載に当たっては、項目の関連性（教育順序）をある程度は考慮して列挙している。ただし、各大学の諸事情に応じて順序を変更することは当然あり得る。また、分野は教育科目を意味するものではない。教育科目への割り当て（すなわち、カリキュラムやシラバス作成）に際しては、必要に応じて分割、統合して割り当てることも可能である。どの学年に、どの分野をどの範囲まで割り当てるか、なども各大学の教育方針やスタッフなどの諸事情に応じて決まるものである。図1に分野、項目等のおおまかな教育順序と関連性を示す。

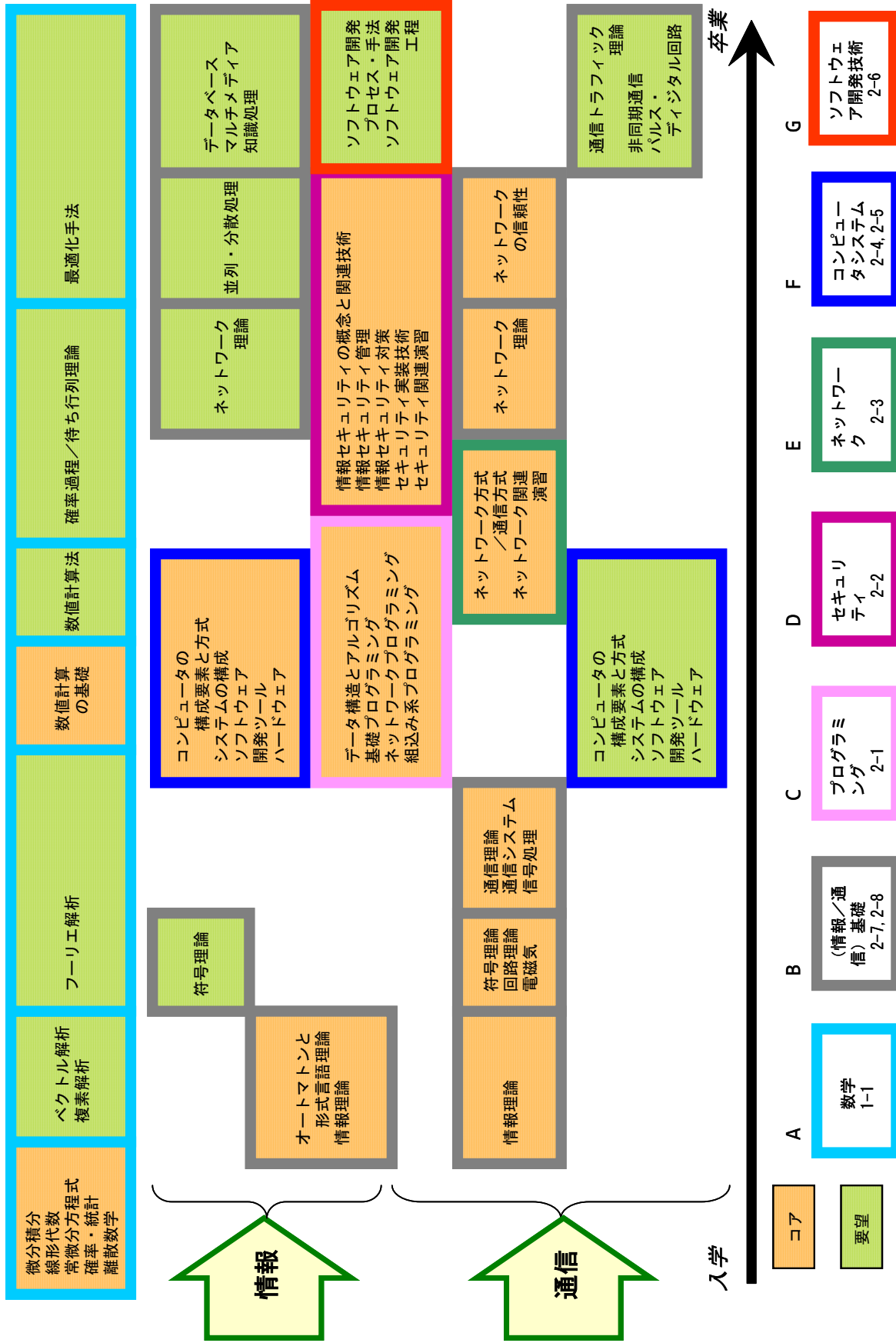


図1 情報・通信分野関連性

【J07 と国際標準化】

情報分野における専門教育のためのカリキュラムとして、情報処理学会、情報処理推進機構（IPA）などが中心となって策定し提案している、情報専門教育カリキュラム標準（J07）がある。これは、IEEE-CS と ACM が共同開発した CC2001-CC2005 を土台とし、かつその国際基準としての整合性を維持しながら日本の情報専門教育状況を反映させている。具体的には、コンピュータエンジニアリング領域（CE）、ソフトウェアエンジニアリング領域（SE）、情報システムエンジニアリング領域（IS）、インフォメーションテクノロジーエンジニアリング領域（IT）、コンピュータ科学エンジニアリング領域（CS）（これら5領域の詳細は後述）の各領域について、カリキュラム標準（知識体系と最低限押さえておくべき項目をコアとして指定したもの）を示したものである。最小限押さえるべき分野（学習域、単元）、項目（トピック）、到達目標、および実施順序などが含まれている。また、時間数や実施年次なども規定した標準カリキュラム案も提示されている。分野、項目、および授業時間数については、日本の現状を考慮して、CC2005 などの指定よりは少なくなっているが、それでもそれぞれの領域について必要な分野、項目が含まれ、そのための授業時間数が割り当てられている。

【情報・通信分野と育成人材像】

本プロジェクトは、学士課程教育における技術者教育のカリキュラム（特にコアカリキュラム）に焦点を当てている。また、「情報・通信分野」は、情報分野と通信分野を対象分野としている。情報・通信分野が想定する育成人材像のイメージを図2に示す。通信分野は、伝達すべき情報の扱い、通信ソフトウェア設計、ネットワーク制御など、情報分野と深く関係する一方で、信号処理、通信理論、電磁波工学、電気・電子機器設計、などといった、必ずしも情報分野と関連性が大きいとは言えない分野も含み、通信分野独自の教育内容を有する。

本プロジェクトでは情報分野の広さも合わせた判断に基づき、情報分野を構成する5領域のうちのCEとSEを中核領域としCSの一部を基礎として含む形で、情報分野と通信分野の共通のコアカリキュラム（情報・通信分野のコアカリキュラム）を構成する。その内容は情報分野に含まれる。策定に当たっては、IPAの情報処理技術者試験の試験要綱を下敷きにし、基本情報技術者試験レベルを原点として、ITパスポート試験レベルおよび応用技術者試験レベルを一部取り込みながら、確かな基礎力と実践力の養成を最優先事項としている。情報処理技術者試験の試験要綱は、J07さらにはCC2005などをベースにしており、実務能力試験範囲としての体裁を整えながらも内容としては国際的標準との整合性が考慮されている。

これに加えて、情報分野としては、IS、ITの一部を取り込んでコアカリキュラムを構成する。一方、通信分野については、上述の独自の教育内容を追加してコアカリキュラムを構成している。図2はこのような人材像のイメージを当該分野との関連性として図示したものである。IS、ITいずれも実務が深く関係し、学士課程教育での実践力養成には重要である。その一方で、その所掌分野が広く、かつ実務システムを扱う必要性が高いため、学士課程教育への取り込みには限界がある。CSは残り4領域の基礎をなす。このような背景から、CE、SEを中核としてコアカリキュラムを構成している。提案コアカリキュラムは、今後（たとえば大学院などで）、CE、IS、SE、IS、IT、CSの各領域あるいは通信分野などにおけるより高度な内容に踏み込んで行くための基礎力養成ベースとなることを意図している。

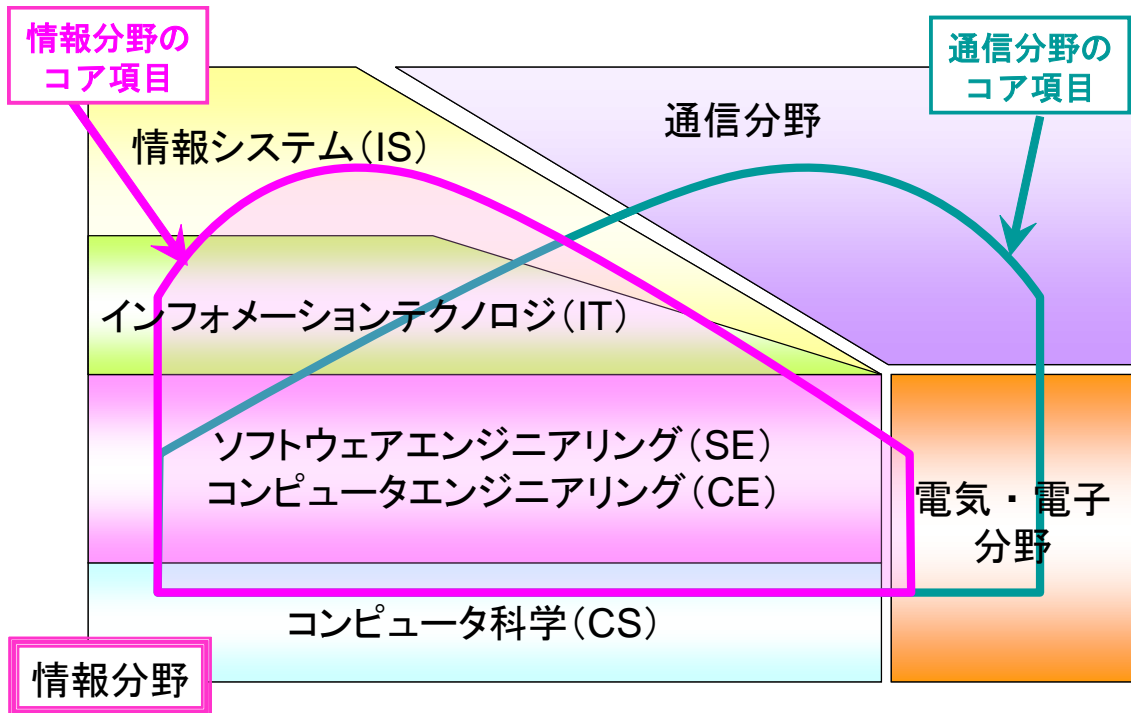


図2 情報専門教育カリキュラム標準 J07(*)と当プロジェクト「情報・通信分野」コアカリキュラムの関連性イメージ
 *：情報処理学会情報処理教育委員会 J07 プロジェクト連絡委員会編

【コアカリキュラムの教育への取込み】

本プロジェクトの目的から、コアカリキュラムとしての分野や項目、および到達目標の設定までを扱い、授業時間や授業実施学年、授業の厳格な実施順序、などまでの設定は記載していない。現状でのカリキュラム策定におけるこれらの設定に際しては、本報告書と共に、必要に応じて上記の CC2005 や J07 などを参考に決定していくことで対応できるものと考えられる。提案するコアカリキュラムは、各大学の現状カリキュラムの調査結果と今後予想される社会情勢を踏まえ、上記人材を育成するコアカリキュラムとしてのあるべき姿を想定して選定されたものである。そのため、その中に記載の分野あるいは項目について、現状では教えていない大学（または、教える環境が整っていない大学）が存在することは十分予想できる。そのような大学には提案するコアカリキュラムを含む方向に進めて行くことを要望するものである。

当プロジェクトは、ここに記載する分野と項目を、いわば技術者教育の情報・通信分野に関するコアカリキュラムとして定着させることを目指している、と言っても過言ではない。JABEE などを「教育プログラムの認定」とすれば、本報告の内容は「具体的な教育コンテンツ認証の基礎」という意味合いを有する。本報告の分野と項目、到達目標などを共通ターゲットとした厳格な評価手法が確立されれば、日本の技術者の情報・通信分野における質保証を強力に後押しすることができるものと考えられる。

【J07 における5領域についての概略説明】

-----コンピュータエンジニアリング領域 (CE: computer engineering) -----

人材の育成方針：組込みスキル標準 (ETSS) (企業が大学教育に期待するスキル分布とほぼ同じ) を持つ技術者。卒業後も持続的に成長していくことができ、かつ技術者としての倫理観を有する人材。

分野：日本の各地域において活躍でき、コンピュータを組込みシステムを応用した製品開発に

用できる能力。単にソフトウェアのみでなくハードウェアの技術も含む。たとえば、通信技術、情報処理技術、制御技術、OSなどのプラットフォーム技術、ソフトウェア開発技術（システム要求定義、ソフトウェア作成、など）など。

-----ソフトウェアエンジニアリング領域（SE: software engineering）-----

人材の育成方針：プログラミング言語の習得にとどまらず、開発ライフサイクルや開発に必要なマネジメントやコミュニケーション、チームダイナミクスなどの能力を身に付ける。情報処理推進機構（IPA）によるITスキル標準（ITSS）、組込みスキル標準（ETSS）、情報システムユーザースキル標準（UISS）などに準拠する。

分野：ソフトウェア構築やソフトウェア設計、検証と妥当性確認、開発マネジメントといったソフトウェアエンジニアリングに必要な実践的技術およびマネジメント能力。

-----情報システム領域（IS: information systems）-----

人材の育成方針：情報システムの作成または活用についての専門知識と能力を持ち、広い視野でシステムをまとめあげることができる人材を育成する。

目的や分野：ISの基礎的な概念を理解すること、ISの学問と研究とは何かを理解すること、IS専門家としての実践的なスキル（技術的な側面と社会的側面）を身に付けること。戦略的要素としてのIS、IS開発の標準、ISの実現とアウトソーシング、知的作業と情報技術、問題解決と経営意志決定、システムとITの概念、組織と情報システム、OSの相互運用とシステム統合、アルゴリズム展開による問題解決、トップダウン実装による問題解決など。

-----インフォメーションテクノロジー領域（IT: information technology）-----

人材の育成方針：情報処理推進機構（IPA）によるITスキル標準（ITSS）の「ITスペシャリスト」「カスタマサービス」「ITサービスマネジメント」、情報システムユーザースキル標準（UISS）の「ISオペレーション」「ISアドミニストレータ」などに、また、情報処理技術者試験のテクニカルエンジニア試験区分「ネットワーク」「システム管理」「情報セキュリティ」などに該当する人材を育成する。

目的や分野：ネットワーク、データベース、セキュリティ、プラットフォーム構築、Web、システム管理やメンテナンス、システムインテグレーションなどの情報技術を身につける。その際には、これらの基礎技術に関する基礎、概念の理解も含まれる。

-----コンピュータ科学領域（CS: computer science）-----

人材の育成方針：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、あるいは様々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、を持った人材の育成。

目的や分野：情報とコンピュータの理論的系統的な扱いを主として、情報の諸分野の基礎としての役割を担う能力を養成する。より具体的な対象分野としては、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、などである。

III. パブリックコメントに対する回答

意見 1 -----

18 【情報・通信分野】 17003

(提出者) K・T

2012-01-04 16:06:17

プログラミング (C)

(2) Cの知識と技術

★手続き型言語を教育する際に、プログラミング言語を特定するような記述の仕方になっているのは、教育機関の自主性を阻害する原因になりやすいので慎むべきだと思います。この節では C 言語を前提とした記述になっていますが、他の言語でも問題ないことを明記し、その上で一つの事例として C 言語の例を挙げるような記述に改めるべきだと思います。

(回答)

C 言語を採用する、という原案は変更しない。

(回答理由)

プログラミング能力を獲得させるために、一つあるいは二つ程度のプログラミング言語に絞って、集中的に取り組むことが効果的だと考えている。プログラミングを、基礎プログラミング、ネットワークプログラミング、組込み系プログラミングと大きく3通りに分けてとらえている。C 言語は、アセンブリ言語に近い機能を持ち、ハードウェアの制御などにも使うことができる。一般的に「C 言語をしっかりとやれば、組込み系プログラミングに通用する」あるいは「C 言語はアセンブラ言語の代用ができる」などと言われるが、これは企業側からの意見としてもよく聞かれる。すなわち、C 言語は、一つの言語で基礎プログラミングと組込み系プログラミングの両方の能力育成に有用な言語である。したがって、プログラミング教育のコアカリキュラムに C 言語を採用することの妥当性は高いと判断できる。また、現在のプログラミング言語でそのような言語は見当たらない。加えて、コアカリキュラムの性質上、多くの選択肢を提示することは避けるべきである。以上の観点から、C 言語以外の選択肢を加えることはしない。(C 言語を凌駕する言語が世に出れば、その時点で見直しを検討すればよい、という前提である。)

(3) Java/C++の知識と技術

★これについても上記と同様、Java ないしは C++は例示であって、他の言語で教育する自由度を認めていることを明記すべきだと思います。

(回答)

指摘を受けて、学習する言語に自由度があることを明示し、かつ候補となる言語を2つから3つに増やすように修正する。(3)の表現についての修正案は以下に★で示す。

(修正理由)

この(3)では、3つのプログラミング言語 Java, C++, Visual Basic .net によって、オブジェクト指向の概念と、言語の特徴を理解し、その記述方法を修得することを目標とする。これらに絞るのは、これら3つ以外の言語(例えば、Ruby や Objective-C など)では、その言語仕様のためすべての目標を達成することが困難であると判断するからである。

=====
★ 修正案

(3) Java/C++/Visual Basic .net の知識と技術

Java, C++, Visual Basic .net のいずれの言語を用いても良い。

【個別到達目標】

. . .

=====
意見 2-----

23 【情報・通信分野】 17006

(提出者) K・T 2012-01-04 16:16:39

コンピュータシステム情報 (F1)

★情報系のエンジニアに不可欠なソフトウェア工学技術がどこにも含まれていません。

- ・ソフトウェアライフサイクル
- ・要件定義, モデリング
- ・段階的詳細化, モジュール化
- ・構造化プログラミング, Jackson 法
- ・ソフトウェアテスト, レビュー, インспекション

といった項目を追加すべきだと思います。

(回答)

指摘に従って, 以下のような分野と項目を G として最後に追加する。

=====追加予定の分野と項目=====

(一応, 情報分野, 通信分野共に要望分野とすることを前提とする。ただし, コア分野に移すこともある。)

G ソフトウェア開発技術

【ソフトウェア開発プロセス・手法】

ソフトウェア開発手法 (開発工程モデル: ウォーターフォール型, スパイラル型, 形式手法), ソフトウェアライフサイクルプロセス (SLCP) など

【ソフトウェア開発工程】

1. ソフトウェア要件定義

ソフトウェア要件の確立 (機能, 能力, インタフェースほか) など

2. ソフトウェア方式設計・ソフトウェア詳細設計

ソフトウェア構造とコンポーネントの設計, インタフェース設計, ソフトウェアユニットのテストの設計, レビュー (インспекション, ウォークスルー), 各種設計手法 (プロセス中心設計, データ中心設計, 構造化設計, オブジェクト指向設計など), モジュールの設計, デザインパターン など

3. ソフトウェアコード作成及びテスト

ソフトウェアコード作成, デバッグ, テストなど

=====

3. ファイルシステム

★ファイルシステムを取り上げるよりも, データベースを取り上げた方が現実的なソフトウェア開発に即しています。この項目全体をデータベースに差し替え, ファイル管理の話題は「1. オペレーティングシステム」に移動するのが妥当だと思います。

(回答)

修正はせず, このままの表現とする。

(回答理由)

ファイルシステムはコンピュータシステムとしては必要である。また, OS との関連性は大きい, やはり別物として説明しておくべきであると判断する。指摘の通り修正すると, コンピュータシステムの中にデータベースを含ませることになり, 説明箇所として適当ではないと判断する。一方, データベースの重要性は認識しており, (B1 情報基礎の要望分野/要望項目としてではあるが) 別途取り上げている。したがって, 現状のままでよいと判断する。

4. アセンブラ言語とそのプログラミング

★この部分は, プログラミングの中にも含まれており, 内容的に重複します。プログラミングに移動してまとめるのが妥当だと思います。

(回答)

敢えて重複させていたが, こちらは削除し,

0 プログラミング

【基礎プログラミング】

2. プログラム言語

(4) アセンブラ言語とそのプログラミング

のみ残すことにする。

意見 3 -----

31 【情報・通信分野】 17008

(提出者) K・T 2012-01-04 16:31:32

情報基礎 (B1)

★情報分野の到達目標の設定全般に関する意見ですが, 書くべき項目が見当たらないので, こちらに記入します。

情報専門教育に対する到達目標を設定する際には, 情報処理学会の J07 カリキュラム標準や ACM 等が定めた CC2005 等を踏まえるのが国際的な通用性を考慮しても妥当だと思います。これらのカリキュラムでは CS, CE, IS, SE, IT の 5 領域が設定されており, 領域毎に方向性が大幅に異なっていますが, 提示されている到達目標では, そうした違いは考慮されていません。このままでは, 国際的な通用性の面で妥当性を欠く目標設定になってしまうでしょう。

技術者教育 (Engineering Education) が対象範囲であることを考慮しても、CE と SE は対象範囲に含まれます。情報分野における到達目標の設定に当たって、2 領域をどのように考慮したかを説明すべきだと思います。

(回答)

現状の案のままとする。

(回答理由)

情報専門教育カリキュラム標準 (J07) は、情報分野における専門教育のためのカリキュラムとして、情報処理学会、情報処理推進機構 (IPA) などが中心となって策定し提案したものである。これは、IEEE-CS と ACM が共同開発した CC2001-CC2005 を土台とし、かつその国際基準としての整合性を維持しながら日本の情報専門教育状況を反映させて、コンピュータエンジニアリング領域 (CE)、ソフトウェアエンジニアリング領域 (SE)、情報システムエンジニアリング領域 (IS)、インフォメーションテクノロジーエンジニアリング領域 (IT)、コンピュータ科学エンジニアリング領域 (CS) の 5 領域の各領域について、カリキュラム標準 (知識体系と最低限押さえておくべき項目をコアとして指定したもの) を示している。最小限押さえるべき分野 (学習域、単元)、項目 (トピック)、到達目標、および実施順序などが含まれている。また、時間数や実施年次なども規定した標準カリキュラム案も提示されている。分野、項目、および授業時間数については、日本の現状を考慮して、CC2005 などの指定よりは少なくなっている。

本プロジェクトでは情報分野の広さも合わせた判断に基づき、情報分野を構成する 5 領域のうちの CE と SE を中核領域とし CS の一部を基礎として含む形で、情報分野と通信分野の共通のコアカリキュラム (情報・通信分野のコアカリキュラム) を構成する。その内容は情報分野に含まれる。策定に当たっては、IPA の情報処理技術者試験の試験要綱を下敷きにし、基本情報技術者試験レベルを原点として、IT パスポート試験レベルおよび応用技術者試験レベルを一部取り込みながら、確かな基礎力と実践力の養成を最優先事項としている。情報処理技術者試験の試験要綱は、J07 さらには CC2005 などをベースにしており、実務能力試験範囲としての体裁を整えながらも内容としては国際的標準との整合性が考慮されている。

これに加えて、情報分野としては、IS、IT の一部を取り込んでコアカリキュラムを形作る。一方、通信分野については、上述の独自の教育内容を追加してコアカリキュラムを構成している。報告書の図 1 には、このような人材像のイメージを当該分野との関連性として図示している。IS、IT いずれも実務が深く関係し、学士課程教育での実践力養成には重要である。その一方で、その所掌分野が広く、かつ実務システムを扱う必要性が高いため、学士課程教育への取り込みには限界がある。CS は残り 4 領域の基礎をなす。このような背景から、CE、SE を中核としてコアカリキュラムを構成している。提案コアカリキュラムは、今後 (たとえば大学院などで)、CE、IS、SE、IS、IT、CS の各領域あるいは通信分野などにおけるより高度な内容に踏み込んで行くための基礎力養成ベースとなることを意図している。

以上述べたことから、現在までの調査段階では、J07 に起因する修正や新規の追加は特には無い。今後、追加や修正が出る可能性は残っている。少なくとも、選択すべき対象は世界標準である。分野設定や項目選定については、J07 のかなりの部分を含めてはいるものの、本プロジェクトの判断に基づいている。その是非も含め、提示したコアカリキュラム案が国内標準あるいは世界標準として認められるかどうかは、今後の評価を待つしかない。

意見 4

55 【情報・通信分野】 17005

(提出者) S・K 2012-01-04 22:39:18

ネットワーク (E)

★どの科目にコメントして良いか判らないが、情報通信分野に関して、知識優先のカリキュラムになっており育成する人材像が不明確なため、自らの考えて答えを出せる人材を育成するカリキュラムにしていきたい。

例えば、ISDNにBチャンネルが64Kbpsに設定した考え方、現在、インターネットの世界で自由に画像、音声の通信が可能になっている理由など、これからの情報システムを構築する上での知恵の原点になるカラクリが現在の情報システムの中に多くあるはずである。これら、先人の知恵を知り、応用できる力がこれからの人材に必要と考える。

以上の理由から、知識習得も大切であるが、知識習得を優先するカリキュラムから、自ら答えて答えを導ける人材を育成できるカリキュラムが、これからの大学教育に必要と考える。是非、ご検討をお願いしたい。

(回答)

新規に組込むべき具体的な事項はない。

(回答理由)

「自ら答えて答えを導ける人材の育成」は、確かに、カリキュラム作成に際して考慮すべき重要な視点であり、貴重な指摘である。残念ながら、具体的な事項は提示されていない。当カリキュラム作成にあたって、このことは意識している。具体的には、演習を取り入れて、学生自らが考え、手を動かし、行動する、ということへの関わりを持たせる配慮はしている。ただし、これは教育方法にも深く関係しており、本報告書では方法論は扱っていない。方法論の扱いは各大学、教員に大きく依存することであり、大学や教員の独自性などとも関係し、取り扱いの難しい懸案である。今後の課題となる。

意見 5

73 【情報・通信分野】 17009

(提出者) A・T 2012-01-05 17:51:41

通信基礎 (B2)

★回路理論の分布乗数回路と三相交流回路は、「要望分野／要望項目」の方が良い。

(注)「分布乗数回路」は「分布定数回路」に訂正すべきである。

(回答)

分布定数回路は通信分野では必要であるので、そのまま残す。(情報分野には、回路理論は含まれていない。)一方、三相交流回路は、指摘を考慮した結果、削除することとする。

意見 6

87 【情報・通信分野】 17008
(提出者) H・M 2012-01-09 12:02:23

情報基礎 (B1)

学科の中から寄せられた個人の意見を、代わって記載いたします。

★データ処理高速化を目的としたインデックス構造等のアクセスメソッドについての項目は必要ではないかと思う。これは、データ構造とアルゴリズムの上に成り立つ内容で、情報・通信分野の他の項目の理解・応用にもつながるものである。

(回答)

現状のままで修正はしない。

(回答理由)

確かに必要性は認められるが、コアカリキュラムに含ませるほど重要度が高いとは判断できない。探索法として見た場合、2分探索法とハッシュ法がコアとして選ぶ一番手であろう。現在の報告書では、単に「探索」となっているが、内容としては2分探索法とハッシュ法に絞っている。(このことをカッコ書きで明示する予定である。) 多分木探索として、要望項目に採用してもいいかもしれない。一方、データベースの要素技術として見た場合にも、要望項目として採用する可能性はある。いずれにしても、かなり細かい議論でデータ構造やデータベースの基礎知識が必要であり、学士課程のコアカリキュラムに含ませることが適当とは考え難い。

★一方、データベースセキュリティについて、カッコつきで多くの項目を並べているが、データモデル、代数、正規化などに加えてこれらを教授する場合は単に名前とその概要にしか触れられないと思われる。それでは応用も効かず、あまり意味がないと思われる。

(回答)

具体的な方策についての言及が無く、対応できない。

(回答理由)

セキュリティは一般に個別技術的要素が強い分野である。一般論としては、個別的技術を多角的に広い視野から捉える必要がある。「B1 情報基礎のデータベースデータ」で扱っているデータベースセキュリティ以外にも、「D 情報セキュリティ/ネットワークセキュリティ」では、セキュリティをより広く本格的に扱っている。いずれでも、多種多様なセキュリティ技術の概要を、いろいろな視点から捉えることになっている。そこには、ポイントになるセキュリティ技術を理解することも含まれている。

意見 7

88 【情報・通信分野】 17003
(提出者) H・M 2012-01-09 12:05:54

プログラミング (C)

学科の中から寄せられた個人の意見を、代わって記載致します。

【データ構造とアルゴリズム】

★他の記述にも言えることかもしれないが、リストアップされている項目が細かすぎる感がある。例えば(2)アルゴリズムの設計と解析において、Prim法、Kruskal法、Ford-Fulkerson法、Dinic法などが並べられている。しかし、そもそもコア分野において重要なことは、学んだことの応用力をつけることであり、そのためには、少ない項目をより深く演習なども併用して少数の事柄を何度も学習するやり方も必要である。

また、リストアップされている項目は、時代の流れとともに変更する必要があるかもしれないものを挙げるのは得策ではないように思われる。リストアップが必要であれば、これらの項目を挙げる際には「例えば」のような言葉をつけておく方がよいのではないかと思う。

(回答)

変更はしない。

(回答理由)

コアカリキュラムは、どの大学の学士課程教育においても必ず含むべき内容であり、ここでの多様性や曖昧性は極力避けるべきものである。むしろ詳細に分野、項目、到達目標を記述しておくことが望ましい。この観点から詳細に記述している。リストアップしている項目は、現状あるいは近い将来において、知識として持つておくべき内容であり、かつデータ構造と共にアルゴリズムを実装する能力を備えておくべきものである。そのような項目として選択している。時代と共に変化しない普遍的と考えられる事項を選択するのは当然で、それを意識して選んでいる。しかしながら、それでもコアとすべき事項が時間経過と共に変わることは避けられない。そのときには修正して対応することを前提としている。

また、アルゴリズムとしては、応用が広く、かつ概念として重要なものに絞って選んでいる。さらに項目間のつながりにも配慮している。ここで言えば、データ構造との関係などにも考慮している。たとえば、Prim法の高速化には、最小値選択の高速化がポイントであるので、ヒープが重要である。一方、Kruskal法の高速化には、互いに素な集合としての扱いが有用であり、そこには、集合を木として扱い、集合を特定する操作や集合を併合する操作の高速化がポイントとなる。ヒープも互いに素な集合の扱いも、各々はデータ構造として教わる。アルゴリズムの選択には、それが実際のアルゴリズムの高速化にどのように使用されるかを把握させる意図も加わっている。最大流問題はその応用の広さから重要度は高い。アルゴリズムとして重要なものは多数あるが、最も基本的で必須と考えられるFord-Fulkerson法、実用性から判断して最も基本的なDinic法を取りあげて、これらを十分理解し、自在に使いこなせるようになることを目指している。コアカリキュラムとしてFord-Fulkerson法ははずせない。また、実用性を重視する観点からは、Dinic法が基本であると判断する。

したがって、指摘のあった「たとえば」などの表現の追加はしない。
