

## Proposal For A New Non-General Education Course: Object-Oriented Programming and Data Abstraction

### 1. Details

#### a. Course Title

0704.114 Object-Oriented Programming and Data Abstraction.

#### b. Sponsor

Stephen J. Hartley and the Computer Science Department.

#### c. Semester Hours

4 semester hours (lecture with lab). For faculty, the course counts as 5 load hours of teaching.

#### d. Course Level

Freshman.

#### e. Prerequisites

Formal and declared status as a Computer Science major or minor or permission of instructor. C– or better in 0704.113 Introduction to Object-Oriented Programming<sup>1</sup>; or a score of 4 or 5 on the **Java** version of the Advanced Placement Computer Science ‘A’ Exam in high school; or C– or better in 0704.103 Computer Science and Programming and C– or better in 0704.112 Java for Object-Oriented Programmers<sup>2</sup>. 1701.122 Precalculus or the high-school equivalent.

#### f. Time and Scale of Implementation

Effective fall 2003 for all entering Computer Science majors and minors. This is one year before the high schools switch to Java for the Computer Science Advanced Placement ‘A’ and ‘B’ exams, which will be based on Java starting spring 2004.

#### g. Curricular Effect

The course proposed here and the proposed new course 0704.113 Introduction to Object-Oriented Programming<sup>3</sup> will replace the three courses, 0701.102 Introduction to Programming, 0704.103 Computer Science and Programming, and 0701.205 Computer Laboratory Techniques in the Computer Science major and minor degree programs.

---

<sup>1</sup>See separate new course proposal.

<sup>2</sup>See separate new course proposal.

<sup>3</sup>See separate new course proposal.

The courses 0701.102 Introduction to Programming and 0704.103 Computer Science and Programming will continue to be offered and taught by the Computer Science Department in their current forms. 0701.205 Computer Laboratory Techniques, however, will be eliminated as a required course for the Computer Science major and minor<sup>4</sup>.

Currently, Computer Science students with no previous programming experience take Introduction to Programming as a free elective, followed by Computer Science and Programming. The effect of this proposal is that Computer Science students with no previous Java programming experience will take Introduction to Object-Oriented Programming and Object-Oriented Programming and Data Abstraction as required courses for the major and minor. Students with sufficient previous Java programming experience will not be required to take Introduction to Object-Oriented Programming. Sufficient previous Java programming experience will be determined by a score of 4 or 5 on the **Java-based** Computer Science AP ‘A’ Exam in high school. The number of semester hours required for the Computer Science major will increase from 120 to 121.

The reason for not switching the languages used in the Introduction to Programming and Computer Science and Programming courses from Visual Basic/C++ to Java is that the students who will still be taking those courses, students not majoring or minoring in Computer Science, are better served, in the opinion of the Computer Science faculty, by the Visual Basic/C++ languages because those languages are more widely used in these students’ curricular areas (mathematics, business information systems, the sciences, engineering).

Students who take 0704.103 Computer Science and Programming and who later decide to major or minor in Computer Science or who wish to take upper division Computer Science courses will need to take the one semester-hour bridge course 0704.112 Java for Object-Oriented Programmers<sup>5</sup>.

## **h. Adequacy of Resources**

Because Computer Science majors and minors will no longer be taking Introduction to Programming and Computer Science and Programming, fewer sections of those courses will be needed. This, together with the elimination of the Computer Laboratory Techniques course from the major and minor degree programs, will provide adequate faculty, classroom, equipment, and laboratory resources to offer Introduction to Object-Oriented Programming and Object-Oriented Programming and Data Abstraction.

## **i. Library Resources**

Library resources are adequate.

## **2. Rationale**

The College Board has decided to change the language used in the high school computer science advanced placement exams from C++ to (a subset of) Java, effective with the 2003–2004 academic year<sup>6</sup>.

“In 1999, the College Board appointed an AP Computer Science Ad Hoc Committee to address the future of AP Computer Science (APCS). Their charge was to make recommendations to the AP Computer Science Development Committee on the future direction of AP Computer Science, including recommendations for changes in the APCS curriculum, and to explore alternative delivery languages. These recommendations were to be based, in part, on a survey of college

---

<sup>4</sup>See separate curriculum change proposal.

<sup>5</sup>See separate new course proposal.

<sup>6</sup>[http://cbweb2s.collegeboard.org/ap/pdf/adhoc\\_report\\_surveys.pdf](http://cbweb2s.collegeboard.org/ap/pdf/adhoc_report_surveys.pdf)

departments conducted in summer 1999 to see what language was likely to be used in the introductory computer science courses in the next three to four years.

“After four meetings, and after considering the surveys conducted in 1999 and 2000 as well as survey results published in the Journal of Computer Science Education (Stephenson and West 14(1), 14(2), April 2000), the Ad Hoc Committee recommends a change from an object-based to an object-oriented approach to programming. The fundamental difference is an inclusion of inheritance. As a result, we further recommend that the language supporting the curriculum be switched from C++ to Java, beginning with the 2003–2004 academic year.

“There are philosophical and pragmatic reasons for recommending these changes. The philosophical reasons are based on three principles that should be considered in choosing a language to express the concepts found in an introductory computer science curriculum: safety, simplicity, and support for object-orientation. Pragmatic reasons for changing are based primarily on the fact that colleges and universities are adopting new approaches based on the three principles. These new approaches are more natural to express in Java than in C++ for reasons described in the appendix that follows this Executive Summary.”

Although C++ is a very popular object-oriented language, widely used in the software industry, it has many “features” inherited from the C programming language, such as arbitrary pointer arithmetic and lack of array bounds checking, that are deplorable pitfalls for beginning Computer Science students. Thus, the College Board and many undergraduate Computer Science departments around the world have embraced Java as a language better suited for teaching object-oriented programming.

The Computer Science department therefore proposes to switch the language used in its introductory programming courses **for Computer Science majors and minors only** from Visual Basic/C++ to Java, no later than the fall 2003 semester, in order to accommodate the increasing proportion, eventually 100 percent, of entering first year and transfer computer science majors who will have already learned some Java.

### 3. Essence of the Course

#### a. Objectives in Relation to Student Outcomes

Upon completion of Object-Oriented Programming and Data Abstraction, students should be able to

- design class hierarchies;
- design interfaces and abstract classes;
- use containers, collections, and iterators;
- recognize design patterns;
- utilize design tools such as class diagrams, CRC cards, and UML use cases;
- design and implement problem-specific data structures;
- understand virtual machines and intermediate languages;
- design and implement sorting, searching, and recursive algorithms;
- incorporate principles of human-computer interaction into program design;
- design effective graphical user interfaces;

- understand event-driven graphics programming;
- use advanced commands on a UNIX system; and
- apply the principles of software engineering to program design, coding, testing, and maintenance.

## **b. Topical Outline/Content**

This is derived from the *Computing Curriculum 2001: Computer Science* revised draft, December 1, 2001, published by the Joint Task Force on Computing Curricula of the IEEE Computer Society and the Association for Computing Machinery.

Object-Oriented Programming and Data Abstraction.

- Review of object-oriented programming: Object-oriented methodology, object-oriented design; software tools.
- Principles of object-oriented programming: Inheritance; class hierarchies; polymorphism; abstract and interface classes; container/collection classes and iterators.
- Object-oriented design: Concept of design patterns and the use of APIs; modeling tools such as class diagrams, CRC cards, and UML use cases.
- Virtual machines: The concept of a virtual machine; hierarchy of virtual machines; intermediate languages.
- Fundamental computing algorithms: Searching; sorting; introduction to recursive algorithms.
- Fundamental data structures: Built-in, programmer-created, and dynamic data structures.
- Advanced UNIX commands.
- Event-driven programming: Event-handling methods; event propagation; exception handling
- Foundations of human-computer interaction: Human-centered development and evaluation; principles of good design and good designers.
- Fundamental techniques in graphics: Hierarchy of graphics software; using a graphics API.
- Software engineering issues: Tools; processes; requirements; design and testing; design for reuse; risks and liabilities of computer-based systems.

## **c. Evaluation and Grading**

Students will be evaluated by traditional methods such as homework, quizzes, written exams, and programming assignments.

## **d. Course Evaluation**

The courses will be evaluated through customary student evaluations as well as regular departmental review.

## 4. Results of Consultations

### a. Departments and Programs

We solicited letters of consultation from Mathematics, Management Information Systems, and Electrical/Computer Engineering. Those letters are attached.

Date: Fri, 08 Mar 2002 14:10:14 -0500  
From: "Ronald Czocho" <Czocho@rowan.edu>  
To: hartley@elvis.rowan.edu  
Subject: Re: letter of consultation

I have a draft of a letter of consultation, but I thought I would let you know what it says before I send it to you. Let me know how you respond to the following:

Thank you for the opportunity to review the changes you have proposed in the required introductory sequence for CS majors and minors. The two courses that you propose should work well for your majors, but they will pose a problem for other students planning to minor in CS or even just take advanced courses in CS.

If, as you say in your proposal, students from other disciplines will be better served by studying Visual Basic and C++, how can you justify changing the minor, which is designed for these students, so that they do not take these languages. If Java is a better language for learning how to program a computer, then Intro to Programming should use this language as well.

I cannot support your proposed change if these courses will be required for the CS minor. In fact, since the Mathematics department has students who often want to take advanced CS courses as well, if these new courses are prerequisites for upper level courses I will not support the new courses at all.

If a change is necessary at the introductory level, I think all students would benefit from this change.

From hartley Tue Mar 12 12:22:37 2002  
To: "Ronald Czocho" <Czocho@rowan.edu>  
Subject: Re: letter of consultation

I have revised the proposal to address your concerns. We will also propose a new one-credit hour bridge course for students who have taken CS&P in C++ and who want to

- major in computer science
- minor in computer science
- take advanced CS courses (but not major or minor in CS)

In your e-mail you express concern that if we change the intro courses for our majors and minors from C++ to Java, then we should do it for CS&P so that all students get Java's advantages. The computer science faculty feels that C++ in CS&P and Visual Basic in Intro to Programming are the right languages for students taking only those courses and not going further in computer science. The approach, called "objects early,"

that we will take in the new courses is appropriate for CS majors and minors because they will continue on to take software engineering and they need a strong dose of object-oriented design early. Students taking only CS&P and/or Intro to Prog do not need and should not have this 'objects early' approach. My own personal opinion is that to try to do so would be a disaster!

I hope this addresses your concerns. Please let me know. Thanks.

Date: Tue, 12 Mar 2002 12:54:59 -0500  
From: "Ronald Czocho" <Czocho@rowan.edu>  
To: hartley@elvis.rowan.edu  
Subject: Re: letter of consultation

Thank you for your efforts. I too think Visual basic and C++ are more appropriate. I just wanted to make sure our majors were not prohibited from deciding to take a minor or co-major after having taken Intro. Your bridge course takes care of my objection and I'll try to get a positive letter to you about the new proposal soon.

From hartley Wed Apr 3 16:06:15 2002  
To: lee@rowan.edu  
Subject: letter of consultation

We need a letter of consultation from MIS for the following related curriculum proposals. They are available in the openarea.

openarea/cs/hartley/curriculumcommittee/2002

cs1NewCourse.pdf	New course proposal: Introduction to OO Programming
cs1cs2Changes.pdf	Changes to major requirements involving the new courses
cs1cs2Minors.pdf	Changes to minor requirements involving the new courses
J00P.doc	New course proposal: C++ to Java bridge course
cs2NewCourse.pdf	New course proposal: OO Programming and Data Abstraction

The proposals are also accessible over the Web at

<http://users.rowan.edu/~hartley/cs1NewCourse.pdf>  
<http://users.rowan.edu/~hartley/cs1cs2Changes.pdf>  
<http://users.rowan.edu/~hartley/cs1cs2Minors.pdf>  
<http://users.rowan.edu/~hartley/cs2NewCourse.pdf>

Thank you for your help.

From hamilton@rowan.edu Thu Apr 4 13:21:16 2002  
Date: Thu, 04 Apr 2002 13:18:58 -0500  
From: "Diane Hamilton" <hamilton@rowan.edu>  
To: <hartley@elvis.rowan.edu>  
Cc: <eze@groupwise.rowan.edu>, <frimpong@groupwise.rowan.edu>, "Jooh Lee" <lee@groupwise.rowan.edu>, <turnerb@groupwise.rowan.edu>  
Subject: Re: letter of consultation

Is there any chance you would consider allowing an MIS major to enroll in these two new courses as electives AFTER having completed CS&P? (The prerequisite at stated would preclude this. Perhaps there could be some grade restriction, i.e., MIS student earning at least a "B" in CS&P). It could be a valuable sequence of electives for a strong MIS student who is ultimately interested in e-commerce.

From hartley Thu Apr 4 13:32:41 2002  
To: "Diane Hamilton" <hamilton@rowan.edu>  
Subject: Re: letter of consultation  
Cc: davis@rowan.edu

I think we could do that on a space available basis: an MIS student who has taken CS&P and wants to go further would first take the one-credit bridge course 0704.112 Java for Object-Oriented Programmers and then take 0704.114 Object-Oriented-Programming and Data Abstraction. CS&P and the bridge course would be roughly equivalent to 0704.113 Introduction to Object-Oriented Programming.

From hamilton@rowan.edu Thu Apr 4 15:07:14 2002  
Date: Thu, 04 Apr 2002 15:04:42 -0500  
From: "Diane Hamilton" <hamilton@rowan.edu>  
To: <hartley@elvis.rowan.edu>  
Subject: Re: letter of consultation

What do you think the chances are that there would be 3-4 seats available in any given year (that's probably all that would have the interest).

From hartley Thu Apr 4 15:24:48 2002  
To: "Diane Hamilton" <hamilton@rowan.edu>  
Subject: Re: letter of consultation

I think the chances for that are very good. Signing a few override slips is always an option if a class is full. And who wouldn't want a few additional interested, motivated, and skilled students in their class, whatever the student's major?

From hamilton@rowan.edu Mon Apr 8 14:00:45 2002  
Date: Mon, 08 Apr 2002 13:58:16 -0400  
From: "Diane Hamilton" <hamilton@rowan.edu>  
To: <hartley@elvis.rowan.edu>  
Cc: "Daniel Davis" <davis@groupwise.rowan.edu>, <eze@groupwise.rowan.edu>  
Subject: Re: letter of consultation

Jooh Lee (dept chair) has suggested that I reply to your request for consultation (on curriculum changes in process) on behalf of the MIS faculty. We have no problems with your course proposals, in light of your agreeing to admit some of our better programmers, should they be interested in the sequence as you outlined below. We will put a proposal through (probably in the fall) that amends our current list of approved electives, to include the courses as listed below - only for students who attain a "B" or better in Computer Science and Programming. Thanks.

From hartley Mon May 13 15:25:47 2002  
To: schmalzel@rowan.edu  
Subject: Re: consult IOOP, OOPDA  
Cc: polikar@rowan.edu  
Status: R

>Existing major track. The current requirements in the ECE major include:  
>  
> Computer Science and Programming (CS&P),  
> Data Structures for Engineers, and  
> Software Engineering.

>  
>From the current proposals, it appears that this pre-existing sequence  
>will remain available. In particular, it means that CS&P remains as one  
>of the possible prerequisites for Data Structures for Engineers, and  
>Data Structures for Engineers remains as one of the possible  
>prerequisites for Software Engineering. Thus, this track means that  
>no change is required for the ECE program of study.

This is correct.

>Current Minor. For ECE students, the current CS minor requirements --- that  
>are in addition to the standard CS course loading --- include:

>  
> Data Structures and Algorithms  
> (instead of Data Structures for Engineers),  
> Computer Lab Techniques, and  
> Discrete Mathematics.  
>

>This 7-SCH minor sequence will still be available for some time in the  
>future. We need clarification so that we can properly advise students as  
>to the timetable.

Discrete Mathematics is being replaced with Discrete Structures,  
effective fall 2002 or Spring 2003 (I'll have to check which since I  
don't remember).

Other than replacing Discrete Math with Discrete Structures, the above  
minor will remain available for some number of semesters. The key here  
is the Computer Lab Techniques course. Our proposals call for replacing  
CS&P and Computer Lab Techniques (7 credit hours) with IOOP and OOPDA (8  
credit hours) in the major and minor requirements, effective fall 2003  
(CS&P and the 1 credit hour Java "bridge" course can be taken instead  
of IOOP). When fall 2003 arrives, many students will still be eligible  
to follow the pre-fall 2003 requirements: all those students who have  
formally and officially declared their status as a Computer Science  
major or minor prior to fall 2003. As long as any of those students  
need to take Computer Lab Techniques, we will continue to offer it.  
You will need to advise your students who want to minor in Computer  
Science that the requirements will change fall 2003 and that if they  
prefer the pre-fall 2003 requirements, they will need to declare formally  
and officially before fall 2003.

>Updated Minor-Rev A. From your proposal, it appears that the future ECE  
>student who wishes to complete the CS minor could need to take:

>  
> The 1-hour bridge course, 0704.112,  
> Java for Object-Oriented Programmers,  
> Data Structures and Algorithms  
> OOPDA  
> Discrete Mathematics  
>  
> This represents a 9-SCH minor requirement.

This is correct for students who formally and officially declare a  
Computer Science minor fall 2003 or later and who have already taken CS&P  
(other than the aforementioned Discrete Structures change).

>Updated Minor-Rev B. Alternatively, it appears that a future ECE student  
>who wishes to complete the CS minor could also take IOOP instead of  
>CS&P if the ECE program were to allow this substitution:

>  
> IOOP (instead of CS&P)  
> Data Structures and Algorithms  
> OOPDA  
> Discrete Mathematics

If you can leave your ECE major requirements as they are now, then any of your students who declare a CS minor fall 2003 or later will take CS&P, the one credit hour Java "bridge" course, Data Structures and Algorithms, OOPDA, and Discrete Structures.

On the other hand, if you change the ECE major requirements so that IOOP replaces CS&P effective fall 2003 (or so that either CS&P or IOOP can be taken), then ECE students who declare a CS minor fall 2003 or later will take IOOP, OOPDA, Data Structures and Algorithms, and Discrete Structures.

From hartley Fri Dec 13 14:23:32 2002  
To: Czochor@rowan.edu  
Subject: letter of consult for the Java stuff

I know it is a pain in the butt to provide letters of consult, but the Provost is going to hold up the Java stuff until you provide a formal letter of consultation.

I appreciate it and I will do my best to practice the Golden Rule.

## b. Campus Expertise

The sponsor does not know of any expertise in the subject area outside the Computer Science department.

## 5. Additional Information

None.

## 6. Catalog Description

This is derived from the *Computing Curriculum 2001: Computer Science* revised draft, December 1, 2001, published by the Joint Task Force on Computing Curricula of the IEEE Computer Society and the Association for Computing Machinery.

0704.114 Object-Oriented Programming and Data Abstraction. 4 s. h.

(Prerequisites: Formal and declared status as a Computer Science major or minor or permission of instructor. C- or better in 0704.113 Introduction to Object-Oriented Programming; or a score of 4 or 5 on the Computer Science Advanced Placement 'A' Exam; or C- or better in 0704.103 Computer Science and Programming and C- or better in 0704.112 Java for Object-Oriented Programmers. 1701.122 Precalculus or the high-school equivalent.)

Objects and data abstraction. Continues from Introduction to Object-Oriented Programming to the methodology of programming from an object-oriented perspective. Through the study of object design,

this course also introduces the basics of human-computer interfaces, graphics, and the social implications of computing, with an emphasis on software engineering. Includes advanced UNIX commands.