

**EXISTING COURSE ADOPTION PROCEDURE (ECAP)
REVISED PROCEDURE FOR BRINGING COURSES OFF THE MASTER COURSE LIST**

PART 1:

Course number: INFO-I210, INFO-I211
Course title: Information Infrastructure I, II
Credit hours: 4, 4
Division offering: BUSE
Official Description: See sample syllabi
Faculty member submitting this proposal: Weber.

PART 2:

Have you contacted all campuses in the IU system who are currently teaching this course and examined their methods of teaching this course? (You should consult the office of the Vice Chancellor of Academic Affairs for help in contacting campuses and procuring sample syllabi.)

Yes.

Below, please list all those campuses which teach the course in question.

Bloomington, Indianapolis, Kokomo, South Bend, Southeast.

PART 3:

Please provide the following elements to the model framework as required by the University Faculty Council (you may attach additional pages):

(1) Describe the course content and your rationale for bringing it forward.

Content: see sample syllabi.

Rationale: These courses will replace CSCI-C201 and 202 in the B.S. in Business MIS concentration. They have similar content and goals to the CSCI courses, but add a laboratory and an extra hour of credit. Laboratory experience is critical to success for many students learning to program.

(2) Describe the course's placement in the program, including its intended audience and any prerequisites. The courses are the first and second in the MIS concentration.

INFO-I210's only real prerequisite is basic algebra (the course description does list INFO-I201 as a "recommended prerequisite or concurrent" course, but as I201 is currently taught at IU East, it really is inessential for I210). INFO-I211's prerequisite is I210.

In addition to meeting the needs of the MIS concentration, the new courses would support the B.S. in Communication Studies as technical electives; would provide another way to satisfy the informatics requirement for general education; would serve as substitutes for CSCI-C201 and C202 as a requirement for math education majors and as an alternative language requirement for Behavioral and Social Science majors. The courses also integrate with the B.S. in Informatics curriculum, which might exist in the future.

(3) List the anticipated student learning objectives and outcomes.

See the sample syllabi.

(4) Discuss any special features of the course.

None.

(5) Attach a proposed syllabus for the course as it would be taught on the IU East campus.

Two sample syllabi attached.

PART 4:

Representatives of your Division need to examine the sample syllabi and other materials from these campuses and determine whether your approach will be comparable. Please describe below your Division's final determination on this point and provide the chair's signature.

Comment: There is wide variation in the choice of programming languages used in INFO-I210 and I211 on different IU campuses and even on the same campus with different instructors. All of these course offerings, however, share some fundamental language-independent goals, which could be summed up as "learning to program" in one or two useful languages. Those goals are served well on our campus by the choice of Python and Java as the programming languages taught.

I approve the above-mentioned course as following the model framework given by other campuses in the IU system.

Division Chair Date

Sample Syllabus for INFO-I210

Course Description

I210 Information Infrastructure I (4 cr.) Recommended prerequisite or concurrent: INFO I101. The software architecture of information systems. Basic concepts of systems and applications programming. Credit given for only one of the following: INFO I210 or CSCI A201 (IUB) [or CSCI-C201 (IUE)].

Comments: This course is about how to program a computer. Programming requires a language, and the language used in this course is Python. As programming languages go, Python is easy to use, but quite powerful. In addition to the language itself, we prepare the ground for further study of informatics by learning about algorithms (procedures for solving a problem), selected data structures (lists and dictionaries), functions, objects, and principles of software design. Students passing this course will be able to develop simple computer applications themselves using the Python language, and they will be well prepared for learning other programming languages. This course should be illuminating and useful to anyone who is interested in understanding how computers work, at the level of performing instructions as directed by a program.

Textbooks

1. John M. Zelle, *Python Programming: An Introduction to Computer Science*. Franklin, Beedle & Associates, 2004.
2. University of Surrey, "Unix Tutorial for Beginners", <http://www.ee.surrey.ac.uk/Teaching/Unix/>
3. Emacs Tutorial.

Learning Objectives

Information is manipulated by programs running on computers; the ability to create and revise programs is therefore an essential skill in informatics. The discipline of programming also teaches fundamental skills of problem solving and develops our capacity to express ourselves in the form of good programming style. Students will learn:

1. Programming: Input, output, variables, assignment; arithmetic, logic, and string operations; iteration (`for` and `while` statements); selection (`if` statement); objects and classes; files; data collections. (As used in Python, but concepts are applicable to nearly all programming languages.)
2. Design principles: top-down and object-oriented design.
3. Programming style.
4. Unix/Linux operating system: file system, file commands, redirection, pipes, wildcards.
5. Use of a text editor.

Campus learning objectives addressed: depth, expression, problem solving.

Grading

8 lab projects, 50 to 60 points each: 440 points (46%)

Lab attendance is required except for students who have turned in all current assignments.

Best 6 of 8 quizzes, 20 points each: 120 points (12%)

Exams 1 and 2, 100 points each: 200 points (21%)

Final exam: 200 points (21%)

Total: 960 points (100%)

Course Content

(Key: Z = Zelle chapter, U = Unix Tutorial, E = Emacs Tutorial)

Week 1. Z1 Computers and Programs. U1 Introduction, Files.

Week 2. Z2 Writing Simple Programs. U2 More files.

Week 3. Z3 Computing with Numbers. E1 Cursor control, hangs, windows, insert and delete.

Week 4. Z4 Computing with Strings. U3 Redirection and pipes.

Week 5. Z5 Objects and Graphics.

Exam 1: chapters 1-5.

Week 6. Z6 Defining Functions. U4 Wildcards, file names.

Week 7. Z7 Decision Structures. E2 files, buffers, searching.

Week 8-9. Z13A Recursion. Z8 Loop Structures and Booleans. E3 Multiple windows.

Week 10. Z9 Simulation and Design.

Exam 2: chapters 6-9.

Week 11. Z10 Defining Classes. U5 Permissions, jobs, processes.

Week 12. Z11 Data Collections. U6 Useful commands.

Week 13. Z13B Algorithm Design and Computability.

Week 14. Z12 Object-Oriented Design.

Final exam.

This Course at Other IU Campuses

- [Bloomington, Fall 2007, Larry Yaeger](#), with Unix and C.
- Indianapolis, Fall 2007, Jason Sisk, with PHP. Syllabus on file.
- Kokomo, Fall 2007, Andrew Kurtz, with Visual Basic. Cross-listed as CSCI-C297 VB Programming. Syllabus on file.
- [South Bend, sample syllabus \(PDF\)](#), with C++.
- [Southeast, 2005, Ray Wisman](#), with Java. Cross-listed as CSCI-C201.
- Northwest and Fort Wayne do not offer this course.

Sample Syllabus for INFO-I211

Course Description

I211 Information Infrastructure II (4 cr.) P: INFO I210. The systems architecture of distributed applications. Advanced programming, including an introduction to the programming of graphical systems. Credit given for only one of the following: INFO I211, CSCI A202 (IUB), or CSCI C212 (IUB) [or CSCI-C202 (IUE)].

Comments: This course covers Java programming. The prerequisite course at IU East covers Python programming, but knowledge of any high-level programming language should be sufficient preparation.

Textbooks

1. John Lewis and William Loftus, *Java Software Solutions*, 5th edition. Addison Wesley, 2006.
2. University of Surrey, "Unix Tutorial for Beginners",
3. <http://www.ee.surrey.ac.uk/Teaching/Unix/>

Learning Objectives

Students completing this course will understand and be able to apply the fundamentals of designing software systems and developing Java programs in the environment of a Linux/GNU operating system. Specific learning outcomes include:

1. Create and revise Java programs using text input and output, variables, assignment, arithmetic and string operations, classes and objects, inheritance, polymorphism, conditionals, exceptions, loops, arrays, recursion, data collections, graphics, and graphical user interfaces.
2. Use top-down and object-oriented design strategies to solve moderately complex problems.
3. Debug and test programs effectively.
4. Use good programming and documentation style.
5. Advanced Unix/Linux skills: file permissions, shell variables and scripts, building and installing software, documentation (man, info), file archival and compression, and version control.

Campus learning objectives addressed: Depth, expression, critical thinking and problem solving.

Grading

6 to 8 lab assignments: 30%

Lab attendance is required unless the student has turned in all open assignments.

Best 4 of 6 quizzes: 10%

Midterm exam: 30%

Final exam: 30%

Total: 100%

Course Content

(Key: LL = Lewis and Loftus chapter; U = Unix Tutorial; [] = other readings.)

Day 1. LL 1 Introduction. U1-U5 if not previously learned.

Days 2-3. LL 2 Data and Expressions. U6 Useful Commands.

Days 4-5. LL Classes and Objects. U7 Compiling and Installing Packages.

Days 6-7. LL 4 Writing Classes. U8 Variables.

Days 8-10. LL 5 Conditionals and Loops. [Shell Scripts.]

Days 11-13. LL 6 Object-Oriented Design. [Makefiles and Kin.]

Days 14-15. LL 1-6, Review and Midterm Exam.

Days 16-18. LL 7 Arrays. [Version Control: Subversion.]

Days 19-20. LL 8 Inheritance. [Documentation: man and info.]

Days 21-22 LL 9 Polymorphism. [File Archival and Compression Utilities: tar, jar, zip, gzip, bzip2.]

Days 23-24 LL 10 Exceptions.

Days 25-26 LL 11 Recursion.

Days 27 LL 12 Collections.

Days 28 LL 1-12 Review.

LL 1-12 Final Exam.

This Course at Other IU Campuses

- [Bloomington, Fall 2007, Rajarshi Guha](#), with Python. Advanced topics include parsing XML and HTML documents, HTML forms, mod_python, database, regular expressions, web services.
- [Bloomington, Spring 2008, Predrag Radivojac](#), with Matlab. Emphasis on data mining.
- [Bloomington, Spring 2008, Christopher Raphael](#), with R.
- Indianapolis, Spring 2008, Jason Sisk, with PHP. Syllabus on file.
- Kokomo, Spring 2008, Andrew Kurtz, with Java. Cross-listed as C309 Object-Oriented Programming. Syllabus on file.
- [South Bend, sample syllabus \(PDF\)](#), with C++
- [IU Southeast, 1999, Gunsham Manwani](#), with C++; cross-listed as CSCI-C202.
- Northwest and Fort Wayne do not offer this course.

