University of Hawaii Community Colleges
Proposal to Initiate, Modify or Delete a Course

1. Type of Action
   - A. Addition
   - B. Deletion
   - C. Modification: 
     - in credits
     - in title
     - in number or alpha
     - in prerequisites or co-requisites
     - Other: Description

2. New Alpha, Number and Title
   - Same
3. Credits 4 credits

4. Old Alpha, Number and Title
   - ICS 111 Introduction to Computer Science
5. Credits 4 credits

6. New Catalog Description
   This is an introductory course for students intending to major in computer science and requiring a computer programming course. Emphasis will be on problem solving, algorithm/pseudocode development, structured programming, computer language coding, implementation and debugging/testing. Students will develop application programs in an IBM microcomputer/DOS/Windows operating system environment. Students will be taught to develop appropriate programs using accepted standards and methodologies. Actual programming is a part of this course.

7. Select box and type specific information in text box.
   - Prerequisites
   - Corequisites or Recommended Preparation
   - Math 27 or equivalent or consent of instructor

8. Student Contact Hours Per Week
   - Lecture 4
   - Lecture/Lab *
   - Lab *
   - Other (click to specify)

9. Proposed Date of First Offering
   - Semester Fall
   - Year 2002

10. This course is proposed for the Program.

11. This course Makes No Difference in the number of credits required for the program/core.

12. Equivalent or similar courses offered in the UH System:

<table>
<thead>
<tr>
<th>Campus</th>
<th>Alpha, Number, Title</th>
<th>Campus</th>
<th>Alpha, Number, Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>LeewardCC</td>
<td>ICS 111 Introduction to Computer Science 1-4</td>
<td>HonoluluCC</td>
<td>ICS 111 Introduction to Computer Science 1-4</td>
</tr>
<tr>
<td></td>
<td>contact</td>
<td></td>
<td>contact</td>
</tr>
<tr>
<td>UH Manoa</td>
<td>ICS 111 Introduction to Computer Science 1-6</td>
<td>KapiolaniCC</td>
<td>ICS 111 Introduction to Computer Science 1-3</td>
</tr>
<tr>
<td></td>
<td>contact - TAs in Lab</td>
<td></td>
<td>contact</td>
</tr>
</tbody>
</table>

13. This course is (check one and click in appropriate textbox and provide details):
   - Already articulated with
     Provide details of existing or desired articulation (date, college(s), purposes, pre-major, etc.) in this space:
   - Appropriate for Articulation with
     Provide details of existing or desired articulation (date, colleges(s), purposes, pre-major or major, etc.) in this space:
     Although ICS 111 does NOT articulate with the program core, it is a required course in the UH ICS program.
   - Not yet appropriate for Articulation.

14. Reason for Initiating, Modifying or Deleting Courses or Other Pertinent Comment:
   Reduction in hours - Many students and especially students choosing a computer science area of study have computers and Internet connections at home. This decreases the time needed for lab usage at school. When this course was first introduced at WCC, most students did not have computer access outside of the computer lab.

Modification of description - The language taught continually changes.

Requested by: ________________________________
Date: 2/36/2002

Approved by: ________________________________
Date: 4/19/02

Dean of Instruction
Date: 5/4/02

CCCM #6100 (Amended for WCC use October 2001)
Levels of Review of Course Proposal at Windward Community College

Course Alpha, Number, and Title:

<table>
<thead>
<tr>
<th>Signatures</th>
<th>Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Department Area (more than one departmental instructor's signature required)</td>
<td></td>
</tr>
<tr>
<td>[Signature]</td>
<td>2-20-02</td>
</tr>
<tr>
<td>[Signature]</td>
<td>2-26-02</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Department</td>
<td></td>
</tr>
<tr>
<td>[Signature]</td>
<td>2/28/02</td>
</tr>
<tr>
<td>Department Chairperson</td>
<td></td>
</tr>
<tr>
<td>[Signature]</td>
<td></td>
</tr>
<tr>
<td>3. Division</td>
<td></td>
</tr>
<tr>
<td>[Signature]</td>
<td>03/07/02</td>
</tr>
<tr>
<td>4. Curriculum Committee Review</td>
<td></td>
</tr>
<tr>
<td>Approved 8-0</td>
<td></td>
</tr>
<tr>
<td>Reason:</td>
<td></td>
</tr>
<tr>
<td>[Signature]</td>
<td>4/9/02</td>
</tr>
</tbody>
</table>

Curriculum Committee Chairperson

CCC #6100 (Amended for WCC use October 2001)
WCC Form for Course Modifications

Course ICS 111 Introduction to Computer Science I
Submitted by Peggy Regentine
Date March 8, 2002

1. What change is proposed in the course? Provide specific information comparing both the “new” and “old” course.

The type of credits will be changed to comply with other community colleges. We presently have 6 contact hours. The change will show 4 contact hours. The rationale is that most ICS students have access to computers at home and do not need the 6 contact hours.

Old: 3 hrs lecture + 3 hrs lab = 6 contact hrs;
New: 4 hrs lecture

New Description: This is an introductory course for students intending to major in computer science and requiring a computer programming course. Emphasis will be on problem solving, algorithm/pseudocode development, structured programming, computer language coding, implementation and debugging/testing. Students will develop application programs in an IBM microcomputer/DOS/Windows operating system environment. Students will be taught to develop appropriate programs using accepted standards and methodologies. Actual programming is a part of this course.

2. What is the rationale for the change?

The change will reduce the number of hours to comply with other community college's curriculum and will update the description to reflect a generic programming language.

3. Is the change substantive enough to require a change in course identification? If so, explain thoroughly.

No

4. Is the course articulated with any 4-year program? No

If yes, give details of the agreement(s) and explain any impact the proposed modifications may have on articulation.

5. Provide details of any additional staff, equipment, facilities, library/media material, faculty preparation and other financial considerations that would be required to implement this course modification. What has been done to provide for these additional costs? Who will teach the course? Is additional preparation needed?

None

CCCM #6100 (Amended for WCC use October 2001)
Original dated WCC 9/91
University of Hawaii Community Colleges
Proposal to Initiate, Modify or Delete a Course
Course Modification Form – Go to next page for Articulation Form

6. Will this course modification result in any alterations in the number of hours required to attain a certificate or degree? * If yes, provide details and justification for these alterations.

   NO

7. If the course is renumbered to 100 or above, does it meet the criteria for transfer level courses? (Go to next page for transfer course criteria.) *
COURSE OUTLINE    ICS 111       FALL 2001

WCC (1110) TTH 1:30-4:30 classroom Noeau 123

COURSE TITLE: Introduction to Computer Science I (4 credits)
INSTRUCTOR: 
OFFICE: 
PHONE: (please leave a message, if no one answers)
E-MAIL: 
OFFICE HOURS: 

COURSE DESCRIPTION
This is an introductory course for students intending to major in computer science and requiring a computer programming course. Emphasis will be on problem solving, algorithm/pseudocode development, structured programming, computer language coding, implementation and debugging/testing. Students will develop application programs in an IBM microcomputer / DOS / Windows operating system environment. Students will be taught to develop programs using accepted standards and methodologies. Actual programming is a part of this course.

COURSE REQUIREMENTS

PREREQUISITE: MATH 27 or equivalent or instructor’s approval. Prerequisites are enforced and the design of this course presumes that the student is adequately prepared.

TEXT: Java Software Solutions, second edition by John Lewis & William Loftus

SUPPLIES: A minimum of two 3.5" High-Density (HD) IBM-formatted diskettes
Folder for organizing lecture notes, handouts, program assignments, exams, etc.

Lectures. Students are required to attend all lectures. If you miss a lecture, it is your responsibility to obtain missed material from a fellow student; you may pick up missed handouts from the instructor. You may tape record lectures. Office hours are NOT for individual lectures. IT IS YOUR RESPONSIBILITY TO KEEP UP WITH THE CLASS AND TO KEEP IN TOUCH WITH ME.

Workload. The workload for this class varies among students as students differ with regard to work & programming experience, learning attitudes & habits and study skills. You are required to read the textbook. You are advised to preview the materials before classes, attend classes, take lecture notes and review class notes regularly so that your learning can be effective/pleasant and your workload can be even & minimized. Since this is a programming class, you are expected to spend an average of 3-20 hours on each programming assignment and a minimum of 6 hours each week for previewing and reviewing class materials (homework and quizzes are ways to ensure that you do this). In this way, the exams should be relatively easy for you.
**Assistance.** The instructor is available during her office hours or call and schedule a specific appointment if necessary. Please come and see me, or call me or send me e-mail for help regarding any questions that you may have.

**Programming Assignments.** 6 programming assignments will be completed this semester. Programs will be worth varying points, which will be specified in the assignment handouts. You are responsible for all written and oral instructions/requirements for programs. All programming assignments are usually due at the beginning of class on the due date. Please note that "testing and debugging of a program" is a part of the program development life cycle, therefore, you are advised to begin working on assignments as soon as possible. Program due date extensions & changes for the entire class may be given... BUT, please do NOT count on due date changes. It is your responsibility to make sure that you turn in all required work for grading and that your diskettes are in good condition and are virus free.

Unreadable diskettes or programs with compile-time (syntax) errors will receive a grade of zero. Programs which run and produce "minor" errors will receive at most 60% of the total possible points. Programs with correct output will not necessarily earn full credit. Programs will be graded according to overall correctness, internal & external documentation and style & structure. It is your responsibility to turn in organized assignments, you may use an envelope (to be provided by the instructor) to turn in work BUT please do NOT use staples.

If a programming assignment is not turned in at the required time on the due date, it is considered LATE and will lose points due to lateness plus it will lose points for errors according to the following:
- 20% of total possible points deducted, if turned in on due date but after required time;
- 50% of total possible points deducted, if turned in after & up to one week of the due date.

Identical or extremely similar programs which result from copying/cheating may receive a failing grade (F) for this course. Programming should be your own effort and work. Refusing to develop programs for you is considered cheating. Catalog.

**Exams.** All exams will be closed book, closed notes. You will be allowed a make-up exam only if you contact me before or on the day of the missed exam and only if you have a documented emergency and can take the exam before it is graded and returned to the other students. Do NOT cheat on exams. For the first cheating act, the parties involved will receive a grade of zero for the exam. For further cheating, the parties involved may receive a grade of F for this course and the instructor retains the option of pursuing academic dishonesty punishment under the university's student conduct code policy. There will be three exams during the semester and all 3 exam scores will be counted toward your total points for the semester.

**Homework & In-class-work Assignments.** Homework/In-class-work may be assigned at various points to highlight key or more difficult concepts to enhance your class learning. You may be assigned to a group for accomplishing an assignment cooperatively when a more interactive learning process is deemed
beneficial. All homework assignments are due at the beginning of class on the due date. All in-classwork assignments are due at the end of class on the assigned day. NO late assignments will be accepted.
Quizzes. Quizzes may be given at various points to ensure that you preview class materials, review lecture notes and attend class regularly for learning effectively. Students who have attended the lectures, read & studied their text & class notes and have successfully completed the assignments should do well. NO make-up quizzes will be given.

Participation/Attendance. You are required to attend class and participate in class discussions during lectures. You may be asked to collect information for class discussion. You are encouraged to foster an active classroom learning environment. This is not only through both the quantity and quality of your questions/answers, but also the professional attitude that you bring into the classroom. Although attendance will not be taken, your grade for assignments and quizzes will suffer if you are absent. You are expected to behave professionally and courteously in the classroom and while a student in this course (i.e., whining is considered extremely unprofessional). You will be asked to leave the classroom if you cause a disruption/disturbance during the class. NO eating or drinking in the classroom is allowed.

EVALUATION/GRADING:

<table>
<thead>
<tr>
<th>Component</th>
<th>Points Possible</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 Programming Assignments</td>
<td>400</td>
</tr>
<tr>
<td>3 Exams (150 pts each)</td>
<td>450</td>
</tr>
<tr>
<td>Homework/Quizzes</td>
<td>150</td>
</tr>
<tr>
<td><strong>TOTAL Points Possible</strong></td>
<td><strong>1000 POINTS</strong></td>
</tr>
</tbody>
</table>

COURSE GRADE COMPUTATION: [standard grading scale]

- A = 90 - 100% 900 - 1000 points
- B = 80 - <90% 800 - 899 points
- C = 70 - <80% 700 - 799 points
- D = 60 - <70% 600 - 699 points
- F = 0 - <60% 0 - 599 points

The "N" grade will NOT be used.

*To obtain a course grade of A or B, you must have turned in ALL the programming assignments and homework AND average at least 70% on the exams. Failure to do so, will drop your course grade one letter grade.*

As much as possible, we will try to follow the grading scheme above, however, this should be considered a tentative scheme and may be subject to change and adjustments.