UC 12299 135

CBM003 ADD/CHANGE FORM

APPROVED APR 24 2013

☒ Undergraduate Council ☐ New Course ☒ Course Change Core Category: Math/Reason Effective Fall 2013	or	Graduate/Professional Studies Council ☐ New Course ☐ Course Change Effective Fall 2013

1. Department: Computer Science College: NSM
2. Faculty Contact Person: Shishir Shah Telephone: 743-3360 Email: sshah@central.uh.edu
3. Course Information on New/Revised course:
   - Instructional Area / Course Number / Long Course Title:
     COSC / 1306 / Computer Science and Programming
   - Instructional Area / Course Number / Short Course Title (30 characters max.)
     COSC / 1306 / COMP SCIENCE AND PROGRAMMING
   - SCH: 3.00 Level: __ CIP Code: 11.0101.00 02 Lect Hrs: 3 Lab Hrs: 0
4. Justification for adding/changing course: **To meet core curriculum requirements**
5. Was the proposed/revised course previously offered as a special topics course? ☐ Yes ☒ No
   If Yes, please complete:
   - Instructional Area / Course Number / Long Course Title:
     ____ / ____ / ____
   - Course ID: _____ Effective Date (currently active row): _____
6. Authorized Degree Program(s): _____
   - Does this course affect major/minor requirements in the College/Department? ☐ Yes ☒ No
   - Does this course affect major/minor requirements in other Colleges/Departments? ☐ Yes ☒ No
   - Can the course be repeated for credit? ☐ Yes ☒ No (if yes, include in course description)
7. Grade Option: Letter (A, B, C ...) Instruction Type: lecture ONLY (Note: Lect/Lab info. must match item 3, above.)
8. If this form involves a change to an existing course, please obtain the following information from the course inventory: Instructional Area / Course Number / Long Course Title
   COSC / 1306 / Computer Science and Programming
   - Course ID: 16778 Effective Date (currently active row): 20120827
9. Proposed Catalog Description: (If there are no prerequisites, type in "none".)
   Cr: 3. (3-0). Prerequisites: MATH 1310 or equivalent. Description (30 words max.): May not be applied to a major or minor in computer science. Overview of basic hardware and software concepts of a computer with design, analysis and programming of efficient algorithms to solve computational problems.
10. Dean's Signature: _____ Date: _____

    Print/Type Name: Ian Evans

REQUEST FOR COURSES IN THE CORE CURRICULUM

Originating Department or College: Computer Science

Person Making Request: Shishir Shah Telephone: 3-3360

Email: sshah@central.uh.edu

Dean's Signature: _____ Date: 3/1/2013

Course Number and Title: COSC 1306, Computer Science and Programming

Please attach in separate documents:

X Completed CBM003 Add/Change Form with Catalog Description

X Syllabus

List the student learning outcomes for the course (Statements of what students will know and be able to do as a result of taking this course. See appended hints for constructing these statements):

Student will develop an understanding of:

1. What are the hardware and software components of a computer system?
2. How do we have to specify solutions to problems if we want computers to do the work?
3. How is it that computers manipulate not just numbers but also things like all the text out there on the Web?
4. If computers are so fast, why are some problems still hard to solve?
5. Are there provable limits to what we can compute? We're asking about problems that no cloud computing or quantum computing will ever be able to solve.
6. Will computers ever rival humans at what we do best? Are they getting close even now?

Component Area for which the course is being proposed (check one):

***Note:** If you check the Component Area Option, you would need to also check a Foundational Component Area.

☐ Communication	☐ American History
X Mathematics	☐ Government/Political Science
☐ Language, Philosophy, & Culture	☐ Social & Behavioral Science
☐ Creative Arts	X Component Area Option
☐ Life & Physical Sciences	MATH REASONING

Competency areas addressed by the course (refer to appended chart for competencies that are required and optional in each component area):

X Critical Thinking ☐ Teamwork

X Communication Skills ☐ Social Responsibility

X Empirical & Quantitative Skills ☐ Personal Responsibility

Because we will be assessing student learning outcomes across multiple core courses, assessments assigned in your course must include assessments of the core competencies. For each competency checked above, indicated the specific course assignment(s) which, when completed by students, will provide evidence of the competency. Provide detailed information, such as copies of the paper or project assignment, copies of individual test items, etc. A single assignment may be used to provide data for multiple competencies.

Critical Thinking:
Questions on exams and homework assignments will assess critical thinking (see attached syllabus with example questions).

Communication Skills:
Students are required to write a paper describing the choice of tools and technologies used in creating their website.

Empirical & Quantitative Skills:
Numerous exam and assignment questions will assess empirical and quantitative skills (see attached syllabus with examples questions).

Teamwork:
n/a

Social Responsibility:
n/a

Personal Responsibility:
n/a

Will the syllabus vary across multiple section of the course? ☐ Yes X No

If yes, list the assignments that will be constant across sections:

Click here to enter text.

Inclusion in the core is contingent upon the course being offered and taught at least once every other academic year. Courses will be reviewed for renewal every 5 years.

The department understands that instructors will be expected to provide student work and to participate in university-wide assessments of student work. This could include, but may not be limited to, designing instruments such as rubrics, and scoring work by students in this or other courses. In addition, instructors of core courses may be asked to include brief assessment activities in their course.

Dept. Signature: _____

University of Houston
Department of Computer Science

Course Syllabus
COSC 1306
Computer Science and Programming

Course Description:

The course will attempt to define computer literacy for the twenty-first century. In this class we'll look at computer systems from the ground up. We'll see how they work and what we can all do to exploit them more effectively. We will focus on the principles of computer science and learn to develop efficient algorithms for a wide variety of problems.

The course will survey the basic hardware and software components of a computer system and introduce the students to algorithmic thinking. The programming assignments will be in a modern scripting language, such as Python or Ruby, in order to relieve the beginner from the drudgery of conventional programming languages.

Course Goals:

Student will develop an understanding of:

- What are the hardware and software components of a computer system?
- How do we have to specify solutions to problems if we want computers to do the work?
- How is it that computers manipulate not just numbers but also things like all the text out there on the Web?
- If computers are so fast, why are some problems still hard to solve?
- Are there provable limits to what we can compute? We're asking about problems that no cloud computing or quantum computing will ever be able to solve.
- Will computers ever rival humans at what we do best? Are they getting close even now?

What to expect:

- Come to class. There will be lectures with demonstrations and sessions where we work problems.
- Work homework problems that reinforce what we've talked about in class.
- Write four or five short programs.
- Build a simple website about yourself and your interests; write a discussion report about your choice of tools and technologies and the reasoning behind your choice.

Required Textbook: Nell Dale and John Lewis, Computer Science Illuminated, Fifth Edition.

Course Grade:

Grade will be based on one midterm exam (20 percent of total grade), one final exam (20 percent of total grade), four individual assignments (10 percent each), and a comprehensive written report and discussion on your choice of tools and technologies used for the design of your website (20 percent of total grade).

Example Questions related to Critical Thinking

1. For the next 48 hours, keep a log of your computing activity. If you do some particular thing many times, it's fine to list it once and just indicate how often or for how many hours you did it. Remember, we're not just talking about interacting with "computers". If it's not a person or animal and it computes, we want to know about it.

We will collect your logs at the beginning of class. We'll ask a few of you to tell us your favorite examples of "computing where you'd least have expected it". We will ask you to describe which of the computing components could be considered as hardware and which as software.

2. For your first program, you should write the "Hello You" program. Hello You should print:

    Hello <your name>. Welcome!

Your program does not need to read any input. You will hardwire your name (in any form you like) into your program.

You should start by creating a new module that contains the code from the one handed out in class. Then modify it so that it produces the output you wish it to generate.

At a minimum, your program should print out a single line with your name. But you are welcome to use some of the techniques in Chapter 2 of your text to print a more interesting display.

3. Memory gets smaller and cheaper every year. One way to see how that has happened over the last decade or so is to look at the history of the iPod. Plot the amount of memory available on the highest end (for purposes of this discussion, the one with the largest memory) model iPod for each year since it came out in 2001. You should be able to get all the data you need at this site, but if some values are missing you can track them down if you know the model names that were available that year: http://www.ipodhistory.com/.

4. If computer A has a 400MB hard drive and computer B has a 20GB hard drive, how much more space does B's drive have than A's? (State this as B has x times as much space as A does. What is x?)

5. Hard drives are slow and prone to breaking (since they are mechanical). So why do computers have them?

6. Suppose that you have a monitor with resolution of 1280 x 800 pixels. It uses 24-bit color. How many bytes are required to store one screen image?

7. We described clock speeds in GHz (or giga hertz).

a. What is the origin of the unit name hertz? (Hint, use Google to find out. While you're at it, use Boolean operators to get rid of all the car rental information from the answer page that Google returns.)
b. Give two examples of things other than computer clocks that are measured in hertz.

Example Questions related to Empirical and Quantitative Skills

1. Convert each of the following decimal numbers to binary:

a. 46

b. 83

c. 467

2. Convert each of the following binary numbers to decimal:

a. 11101

b. 110111

3. Convert each of the following decimal numbers to hex (Hint: Convert to binary first.):

a. 159

b. 234

4. Convert each of the following hex numbers to decimal:

a. B8

b. 2E

5. Show the result of each of the following binary arithmetic operations:

a. 111101 + 111 =

b. 1101 – 11 =

c. 10101 * 111 =

6. The following string corresponds to the hex description of the ASCII encoding of an English sentence. What is the sentence? (You can find the ASCII equivalence table here: http://www.asciitable.com/ or in my Encodings slides.)

4465657020426C756520776F6E20696E20313939372E