New Course Approval Cover Form
Montana State University

This four-page form collects basic information about the proposed new course, provides information on the approval process, and includes all required approvals. Additional information (see INFO sheet) is also required as part of the New Course Packet.

Proposed New Course Information

Requested Rubric, Course Number, Core Designation (if needed): CSCI 107
Example: PHL 361 RH

Course Title: The Joy and Beauty of Computing
Abbreviated Course Title (≤ 30 chars): Joy and Beauty of Computing
First Semester to be Offered: Fall 2013
Submitted by: John Paxton
Submitter’s Contact Info: Phone, Email: x5979 paxton@cs.montana.edu
Instructor: John Paxton
Department: Computer Science
College: Engineering

New Course Review Process

Instructor completes the New Course Packet, with Core information if a Core designation is requested.
Instructor checks for “equivalent” course in the MUS system and recommends a common or unique course number.

Department Head’s signature indicates that course has been approved by the process used within the Department.
The Chair of the College Curriculum Committee signs to indicate College academic approval.
The College Dean signs to indicate that adequate resources are available to offer the course. Supporting Information (Dean’s Statement) is typically required.
The New Course Packet (as PDF) is uploaded to the Provost’s Office server for distribution to other committees.

Course requests are sent to Curriculum and Program Committee (CPC). Core reviews are sent to appropriate Core subcommittee. Committees work in parallel when possible to speed approval process. Special topics courses (291, 491) skip the CPC review (limited to two years.)
Provost’s Office reviews the new course request. New courses are submitted to MUS for Common Course Number (CCN) review. Dean and Department informed upon approval.

Approved new course sent to Registrar for inclusion in the Catalog and Schedule of Classes

APPROVALS

John Paxton 8/30/12
Submitter

John Paxton 8/30/12
Department Head

Mahtab Hourani 9/10/12
Chair, College Curriculum Comm.

Mahtab Hourani 9/10/12
Dean

Chair, Core Subcommittee (if app.)

Date

Chair, CPC

Date

Assoc. Provost

Date

Note: This diagram illustrates the typical flow path, but at any review step there can be a request for additional information or modifications. Careful review in early steps is the best way to speed the overall process.
INFORMATION NEEDED FOR COMMON COURSE NUMBERING

The process for identifying a common course number for a new course is as follows:

1. Course learning outcomes are prepared for the new course.
2. The person submitting the new course request looks at the CCN website to see if a course with similar outcomes already exists in the MUS system.
   
   http://www.mus.edu/Qtools/CCN/ccn_default.asp
   
   • If a course exists with at least 70% of the same outcomes, the course is considered “equivalent” to the proposed new course, and the new course should use the existing rubric and course number.
   • If no “equivalent” course is found, the person submitting the new course request should identify a unique course number that has not been used by any other course in the MUS system.
3. The requested rubric and course number are submitted as part of the new course paperwork.
4. The Provost’s Office submits the learning outcomes and the requested rubric and course number to the MUS to have a course number assigned to the course. (This will typically be the requested course number, but it could be changed.)
5. The assigned common course number is reported back to the person submitting the new course request.

<table>
<thead>
<tr>
<th>Requested Rubric, Course Number, Core Designation (if needed):</th>
<th>CSCI 107</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Title:</td>
<td>The Joy and Beauty of Computing</td>
</tr>
<tr>
<td>Credits:</td>
<td>3</td>
</tr>
<tr>
<td>Department Offering Course:</td>
<td>Computer Science</td>
</tr>
<tr>
<td>College:</td>
<td>Engineering</td>
</tr>
</tbody>
</table>

Is there an “equivalent” course in the MUS System?:  
☐ Yes  ☑ No

Learning Outcomes for the proposed new course:

- Appreciate the variety of ways in which computing can improve the human condition.
- Appreciate how computational skills can benefit one's career and life.
- Design and implement short programs in an interpreted language such as Python.
- Understand some of the significant ideas and people that underlie computing's past, present and future.
CSCI 107
The Joy and Beauty of Computing
Joy and Beauty of Computing
Fall 2013
8467
Computer Science
Engineering

Is the requested course number available? (x4155 to check)
[ ] Yes  [ ] No
Frequency of course offering:
[ ] Annually  [ ] Alternate Years, starting ______
Semester(s) offered (check all that apply):
[ ] Summer  [ ] Fall  [ ] Spring
Summer Options (check all that apply):
[ ] First 6 weeks  [ ] Second 6 weeks  [ ] 12 weeks

Credits by mode of instruction:
- Lecture: __________
- Seminar: ________
- Independent Study: ________
- Lab/Studio: ________
- Recitation/Discussion: ________
TOTAL CREDITS: __________

Primary Mode of Delivery:
[ ] Face-to-face  [ ] Internet/Web-based
[ ] Interactive Video  [ ] Audio and/or Video Tape

Time and Location — Call the Registrar's Office at x4155 to find a time and location for the course.

<table>
<thead>
<tr>
<th>Assigned Day(s):</th>
<th>M</th>
<th>Tu</th>
<th>W</th>
<th>Th</th>
<th>F</th>
<th>Sa</th>
<th>Su</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned Time(s):</td>
<td>1310-1400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assigned Building:</td>
<td>TBA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assigned Room:</td>
<td>TBA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capacity (room capacity, or enrollment “cap”):</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Co- and Pre-Requisites — Courses numbered 200 and above are normally expected to have prerequisites. When listing multiple prerequisites, please separate courses with “and” if both are required, or “or” if only one is required.

Prerequisite(s):
Co-Requisite(s):

Course Description — Provide a course description of 40 words or less for the MSU Catalog.

Examines the computing field and how it impacts the human condition. Introduces exciting ideas and influential people. Provides a gentle introduction to computational thinking using the Python programming language.
DEAN’S STATEMENT

The reviewing committees are being asked to take a closer look at the resources required for each proposed new course. In many cases new courses will replace existing courses and the new course request is effectively resource neutral, however that is not always the case. For example, a new elective course that would result in distributing an existing student population across a larger number of courses would represent a significant increase in expenditures for the new course, and no increase in total student credit hours. A funding mechanism for such a course would need to be identified. The Dean’s Statement is the place to document how the costs of the proposed new course will be covered.

This proposal is revenue neutral. The proposed course (CSCI 107, The Joy and Beauty of Computing) will replace a course that will no longer be offered (CAPP 120, Introduction to Computers). Each course is offered during Fall Semester and the enrollments should be similar. No funding mechanism is needed.
New Undergraduate Course Narrative
Montana State University
Updated August 23, 2012

Please provide the following information in narrative format. Substantive responses to all criteria are required. Although not required, a draft syllabus can also be helpful to the committee in understanding the details of the proposed course.

General Course Information
1. Requested Rubric, Course Number, and Core Designation (if any)
   > CSCI 107

2. Course Title
   > The Joy and Beauty of Computing

3. Provide a general description of the course explaining the need for the course, its goals, and its overall structure. This is the most important part of the application and should offer a good sense of what students will experience by taking this class.
   > The worldwide demand for graduates with computing skills far outstrips the supply. Two reasons for this shortage are (1) the computing field suffers from underrepresentation (it primarily attracts white and Asian men) and (2) what people in the computing field do is poorly understood by the general public. This course is designed to combat these problems by providing students with a motivational introduction to the field of computing. The course has the high level goals of (1) enabling students to see how computing has transformed and improved the human condition and will continue to do so in the future, (2) providing students with computational thinking skills that will serve them well regardless of their major or career and (3) encouraging students to further their computational abilities by pursuing additional computing-related courses, minors, certificates or even majors. The attached syllabus lists the specific course outcomes.

4. Based on what types of student work (e.g., tests, homework assignments, papers, performances, etc.) will grades be determined?
   > Please see the attached syllabus. Grades will be determined based on a midterm, a final, and small programming assignments.

5. Provide a course content outline containing all major topics plus a brief description of the material to be covered under each major topic heading.
   > Please see the attached syllabus.

6. List required texts or other required references.
   > Please see the attached syllabus. The references are available online.
7. What are the estimated enrollment and student credit hour (SCH) production?
   [SCH = (enrollment * credits)]
   
   > 50 * 3 = 150

8. Will there be an enrollment cap that restricts enrollment below the level of student demand? If so, what is the enrollment cap and why is it necessary?
   
   > N/A assuming that our estimate of 50 is roughly accurate.

9. Will course be a "restricted enrollment" course? If so, why is restricted enrollment necessary?
   
   > No.

10. Describe how the success of the course will be evaluated? ("End-of-semester student evaluations" is not the answer to this question. How will the instructor determine if the learning outcomes are being met, and how will the department determine if the course is fulfilling its intended purpose?)

   > In addition to tracking student enrollments and the percentage of students who receive DFWs, the department will also track the number of students who take this course and subsequently take CSCI 111 - the first course for computer science majors and minors. We anticipate that The Joy and Beauty of Computing might serve as a better entry point for students who want to explore the field of computing, but who might not be comfortable starting out with CSCI 111. Even though CSCI 111 is designed to serve all students, it can be intimidating to students when they learn that some of their classmates have been programming since they were 10. I anticipate that a computer science pedagogy paper will result from our experiences with The Joy and Beauty of Computing.

   All graded assignments will be related to the course learning outcomes.

11. Is the instructor a member of the regular faculty (i.e., tenured or tenure-track)? If no, please describe the instructor's qualifications, attach a Vita, and provide a separate letter of support, signed by the department head (or appropriate unit director), addressing the instructor's qualifications to teach this course.

   > Yes.

**Level of Offering**

12. Has the course been offered previously under 280/291 or 480/491? If so, when? Under what number? What was the enrollment? What level of students took the course?

   > No.

13. Justify the level of course offering.

   > The course has no pre-requisites or co-requisites. It provides a student with a high level overview of the field of computing and might motivate students to take additional computing courses. Therefore, a 100 level designation is appropriate.
Relationship to other Courses, Curricula, and Departments

14. Does this course build on or interrelate with other courses in your curriculum or related curricula? If so, which ones?

> Yes. It could serve as a feeder for CSCI 111; the first class that Computer Science majors and minors are required to take. The Joy and Beauty of Computing can better serve students who either (1) want to explore the field of computing, (2) do not have prior programming experience and are intimidated by the fact that some students in CSCI 111 do and/or (3) do not have the proper math background to take CSCI 111. The Joy and Beauty of Computing will allow students to gain confidence in their technical skills while also better understanding the flexibility and variety of computing careers.

15. Do the topics in the proposed course duplicate or reiterate those in other courses in this or any other department? If so, how do the coverage and educational experience differ and how is this duplication or reiteration justified? Also, what liaison (which is expected in cases of apparent overlap) has been conducted with other departments? Report reactions, both favorable and unfavorable.

> Mildly. Part of the course introduces computational thinking using the programming language Python. Because all programming languages feature similar constructs (sequencing, decisions, iteration, etc.), there will be mild overlap with courses that introduce other languages such as CSCI 111 (Java) or CSCI 112 (C).

16. What programs (departments, colleges) will be impacted by the SCH production of this course? That is, where do you think the SCH in the proposed course are likely to come from? If the expected SCH production of the proposed course is greater than 1000, and the SCH are expected to come from other colleges, what steps have been taken to make the other units aware of the potential loss of SCH? Report reactions, both favorable and unfavorable.

> I anticipate that the primary SCH production will come from first-year students who are considering a major or minor in Computer Science, but who do not have prior programming experience. The course should not impact other departments or colleges in any significant manner.

17. If this proposed course has a significant interdisciplinary component, please explain briefly. Otherwise, indicate n/a.

> N/A. However, the course is interdisciplinary with respect to the fact that computing is pervasive in society. As the course explores how computing is being used to solve society’s most pressing problems, students will learn that computing connects naturally with most other disciplines. Furthermore, a computing minor can be an effective complement to any major.

Students Served

18. Does the proposed course serve majors only? Non-majors only? Both majors and non-majors? What other majors might be interested in this course? State areas or disciplines to be served and indicate the specific efforts that will be made to make the course material relevant to all disciplines served.

> The course is designed to serve any student from any major who wants to explore the field of computing. In addition to teaching students how to think computationally using the Python
programming language, the course will also explore computing topics of interest (e.g. Facebook, household robots, autonomous vehicles) that should appeal to a wide variety of students.

Resources
19. What additional resources (e.g., additional instructional FTE, required technologies), if any, will be required to offer this course? Are there any resource issues for the students who will take the course (e.g., required technologies, travel, on-line access requirements)? Will there be an additional fee charged to students taking this course? Please explain.

> No additional resources will be required. We are replacing a course that has a Fall 2012 enrollment of 60 (CAPP 120, Computer Literacy) with this one. As a side note, CAPP 120 is also offered by Gallatin College in the Fall Semester.

20. What existing information resources – print (books, journals, documents), audiovisual (videos, DVDs, CDs or other), and/or electronic (e-books, databases, electronic journals and web sites) – provided by the MSU Libraries will be used by students in this course? Provide examples as well as descriptive information. If additional information resources are necessary, please discuss those acquisitions with the library (x6549 Collection Development) at least three months prior to the beginning of the semester in which this course will be taught.

> None. The course materials are available on the web.

Other Supporting Material
21. Include any additional information you feel is needed to support this request.

> Other universities such as Harvey Mudd and UC Berkeley have found that courses similar to the one being proposed provide students that are historically underrepresented in computer science with a better entry point into a computing field.
# CSCI 107: The Joy and Beauty of Computing

## Fall 2013

<table>
<thead>
<tr>
<th>Date</th>
<th>Python Topic</th>
<th>Computing Topic</th>
<th>Graded Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>08.26</td>
<td>General Introduction</td>
<td>Maria Klawe</td>
<td></td>
</tr>
<tr>
<td>08.28</td>
<td>General Introduction</td>
<td>Improve Urban Infrastructure</td>
<td>Advance Health Informatics</td>
</tr>
<tr>
<td>08.30</td>
<td>General Introduction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>09.02</td>
<td>Labor Day - No Class!</td>
<td>Reverse-Engineer the Brain</td>
<td></td>
</tr>
<tr>
<td>09.04</td>
<td>Simple Python Data</td>
<td>Secure Cyberspace</td>
<td></td>
</tr>
<tr>
<td>09.06</td>
<td>Simple Python Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>09.09</td>
<td>Simple Python Data</td>
<td>Sheryl Sandberg</td>
<td></td>
</tr>
<tr>
<td>09.11</td>
<td>Python Turtle Graphics</td>
<td>Enhance Virtual Reality</td>
<td></td>
</tr>
<tr>
<td>09.13</td>
<td>Python Turtle Graphics</td>
<td>Advance Personalized Learning</td>
<td></td>
</tr>
<tr>
<td>09.16</td>
<td>Python Turtle Graphics</td>
<td>Steve Jobs, Steve Wozniak</td>
<td></td>
</tr>
<tr>
<td>09.18</td>
<td>Python Modules</td>
<td>Apple</td>
<td></td>
</tr>
<tr>
<td>09.20</td>
<td>Python Modules</td>
<td>MSU Computing Degrees</td>
<td></td>
</tr>
<tr>
<td>09.23</td>
<td>Python Modules</td>
<td>Luis von Ahn</td>
<td></td>
</tr>
<tr>
<td>09.25</td>
<td>Functions</td>
<td>Carnegie Mellon</td>
<td></td>
</tr>
<tr>
<td>09.27</td>
<td>Functions</td>
<td>Duolingo</td>
<td></td>
</tr>
<tr>
<td>09.30</td>
<td>Functions</td>
<td>Alan Turing</td>
<td></td>
</tr>
<tr>
<td>10.02</td>
<td>Selection</td>
<td>Turing Test</td>
<td></td>
</tr>
<tr>
<td>10.04</td>
<td>Selection</td>
<td>Technological Singularity</td>
<td></td>
</tr>
<tr>
<td>10.07</td>
<td>Selection</td>
<td>Bill Gates</td>
<td></td>
</tr>
<tr>
<td>10.09</td>
<td>More About Iteration</td>
<td>Microsoft</td>
<td></td>
</tr>
<tr>
<td>10.11</td>
<td>More About Iteration</td>
<td>Tools of Scientific Discovery</td>
<td></td>
</tr>
<tr>
<td>10.14</td>
<td>Review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.16</td>
<td>Midterm Exam</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.18</td>
<td>Return Exams</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.21</td>
<td>More About Iteration</td>
<td>Ada Lovelace</td>
<td></td>
</tr>
<tr>
<td>10.23</td>
<td>Strings</td>
<td>Early History of Computers</td>
<td></td>
</tr>
<tr>
<td>10.25</td>
<td>Strings</td>
<td>Career Opportunities</td>
<td></td>
</tr>
<tr>
<td>10.28</td>
<td>Strings</td>
<td>Grace Hopper</td>
<td></td>
</tr>
<tr>
<td>10.30</td>
<td>Lists</td>
<td>Grace Hopper Conference</td>
<td></td>
</tr>
<tr>
<td>11.01</td>
<td>Lists</td>
<td>Computing in WWII</td>
<td></td>
</tr>
</tbody>
</table>

**Python Program - 10%**
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Instructor</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.04</td>
<td>Lists, Files, Files</td>
<td>Tim Berners-Lee</td>
<td>The Connected World</td>
</tr>
<tr>
<td>11.06</td>
<td></td>
<td></td>
<td>Massive Open Online</td>
</tr>
<tr>
<td>11.08</td>
<td></td>
<td></td>
<td>Courses</td>
</tr>
<tr>
<td>11.11</td>
<td>Veteran's Day - No Class!</td>
<td></td>
<td>Autonomous Vehicles</td>
</tr>
<tr>
<td>11.13</td>
<td>Files, Dictionaries</td>
<td></td>
<td>Household Robots</td>
</tr>
<tr>
<td>11.15</td>
<td></td>
<td></td>
<td>Python Program -</td>
</tr>
<tr>
<td>11.18</td>
<td>Dictionaries</td>
<td>Richard Stallman</td>
<td>10%</td>
</tr>
<tr>
<td>11.20</td>
<td>Thanksgiving - No Class!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.22</td>
<td>Thanksgiving - No Class!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.25</td>
<td>Dictionaries, Recursion</td>
<td>Larry Page, Sergey Brin</td>
<td></td>
</tr>
<tr>
<td>11.27</td>
<td>Recursion</td>
<td>Google</td>
<td></td>
</tr>
<tr>
<td>11.29</td>
<td></td>
<td>Social Issues Involving</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Google</td>
<td></td>
</tr>
<tr>
<td>12.02</td>
<td>Recursion, Review</td>
<td>Mark Zuckerberg</td>
<td></td>
</tr>
<tr>
<td>12.04</td>
<td>Review</td>
<td>Facebook</td>
<td></td>
</tr>
<tr>
<td>12.06</td>
<td></td>
<td>Social Issues Involving</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Facebook</td>
<td></td>
</tr>
<tr>
<td>Week of</td>
<td>Final</td>
<td></td>
<td>Final - 25%</td>
</tr>
<tr>
<td>12.09</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Textbook**

- *Learning with Python: Interactive Edition (Using Python 3.x)*

**Instructor**

- John Paxton

**Course Outcomes**

At the end of the course, students should be able to

- Appreciate the variety of ways in which computing can improve the human condition.
- Appreciate how computational skills can benefit one's career and life.
- Design and implement short programs in an interpreted language such as Python.
- Understand some of the significant ideas and people that underlie computing's past, present and future.
Other Ideas

- Visit the American Computer Museum
- Watch the Codebreaker documentary
- Watch Randy Pausch's Last Lecture
- Learn about Robin Karp or Linus Torvalds
- Watch Doug Engelbart's Mother of All Demos

Grading

- Midterm - 15%
- Final - 25%
- Programming Assignments using Python - 60%

Last modified: August 29, 2012.