Date of request: 08-OCT-2019

Request ID: A-2019-CS-31

College: A

Department: CS

Initiator name: Jason Johnson

Initiator email: jason.e.johnson@wmich.edu

Proposed effective term: 202040

Does course need General Education approval?: N

Will course be used in teacher education?: N

If 5000 level course, prerequisites apply to: U

Proposed course data:
New Course CS 1210
New course selected: This new course is not seeking approval as a general education course.

1. Proposed course prefix and number:
CS 1210

2. Proposed credit hours:
1

3. Proposed course title:
Introduction to C

4. Proposed course prerequisites:
CS 1120

5. Proposed course corequisites:
none

6. Proposed course prerequisites that may be taken concurrently (before or at the same time):
none

7. Minimum grade for prerequisites (default grades are D for Undergrad and C for Grad):
C

8. Major and/or minor restrictions:
Not Applicable
9. List all the four-digit major and/or minor codes (from Banner) that are to be included or excluded:
none

10. Classification restrictions:
Not Applicable

11. List all the classifications (freshman, sophomore, junior, senior) that are to be included or excluded:
none

12. Level restriction:
Not Applicable

13. List the level (undergraduate, graduate) that is to be included or excluded.
Not Applicable

14. Do prerequisites and corequisites for 5000-level courses apply to undergraduates, graduates, or both?
Not Applicable

15. Is this a multi-topic course?
No

16. Proposed course title to be entered in Banner:
Introduction to C

17. Is this course repeatable for credit?
No

18. Is this course mandatory credit/no credit?
No

19. Select class type:
Lecture/Lab/Discussion

20. How many contact hours per week for this course?
1

A. Please choose Yes or No to indicate if this class is a Teacher Education class:
No

B. Please choose the applicable class level:
Undergraduate

C. Please respond Yes if this is a current general education course and/or a course being submitted for the new WMU Essential Studies program. Please respond No if it is neither.
No
D. Explain briefly and clearly the proposed improvement.
Create an undergraduate level, online course in the fundamentals of the C programming language.

E. Rationale. Give your reason(s) for the proposed improvement. (If your proposal includes prerequisites, justify those, too.).
Students often report difficulty adapting to the C language when entering classes such as CS 2240 and CS 3240 which require the use of the C language. This course will give students an introduction to C so that they will have the opportunity to learn how to program in C before entering classes that require its use.

CS 1120 is a required prerequisite because this class will be an introduction to the C language, but not to programming itself. Students will need to be competent programmers to succeed in this course, so success in CS 1120 will be required.

F. List the student learning outcomes for the proposed course or the revised or proposed major, minor, or concentration. These are the outcomes that the department will use for future assessments of the course or program.
Students will be able to:
• Design, implement, test, debug, and document programs in C.
• Write simple and intermediate-level C programs that compile and run correctly.
• Demonstrate how to write and use functions, how the stack is used to implement function calls, and parameter passing options.
• Demonstrate use of use common data structures typically found in C programs such as arrays and structs.
• Write programs that perform explicit memory management.
• Use pointers effectively; understand the risks of memory leaks and strategies for minimizing such risks.

G. Describe how this curriculum change is a response to student learning assessment outcomes that are part of a departmental or college assessment plan or informal assessment activities.
This change is being made in response to feedback from students who find it difficult to pick up the C language without formal instruction. The lack of such instruction negatively affects their performance in later required classes, so this change is being proposed to satisfy the need pointed out by student feedback.

H. Effect on other colleges, departments or programs. If consultation with others is required, attach evidence of consultation and support. If objections have been raised, document the resolution. Demonstrate that the program you propose is not a duplication of an existing one.
None.

I. Effect on your department’s programs. Show how the proposed change fits with other departmental offerings.
This new course will help students focus on the content of later classes rather than trying to cope with a new language as well as learn the course content. This will help our students to
succeed in future classes such as CS 3240.

J. Effects on enrolled students: are program conflicts avoided? Will your proposal make it easier or harder for students to meet graduation requirements? Can students complete the program in a reasonable time? Show that you have considered scheduling needs and demands on students' time. If a required course will be offered during summer only, provide a rationale.

While students will be required to take this course, which has not been the case in the past, it will be offered in both Fall and Spring semesters each year. That being the case, scheduling will not be difficult and the time required to complete a Bachelor of Science in Computer Science degree will not change.

K. Student or external market demand. What is your anticipated student audience? What evidence of student or market demand or need exists? What is the estimated enrollment? What other factors make your proposal beneficial to students?

Estimated audience is all undergraduate CS students.

L. Effects on resources. Explain how your proposal would affect department and University resources, including faculty, equipment, space, technology, and library holdings. Tell how you will staff additions to the program. If more advising will be needed, how will you provide for it? How often will course(s) be offered? What will be the initial one-time costs and the ongoing base-funding costs for the proposed program? (Attach additional pages, as necessary.)

Offering this class will not require any additional faculty or advising resources.

M. With the change from General Education to WMU Essential Studies, this question is no longer used.

For courses requesting approval as a WMU Essential Studies course, a syllabus identifying the student learning outcomes and an action plan for assessing the student learning outcomes must be attached in the Banner Workflow system.

Not Applicable

N. (Undergraduate proposals only) Describe, in detail, how this curriculum change affects transfer articulation for Michigan community colleges. For course changes, include detail on necessary changes to transfer articulation from Michigan community college courses. For new majors or minors, describe transfer guidelines to be developed with Michigan community colleges. For revisions to majors or minors, describe necessary revisions to Michigan community college guidelines. Department chairs should seek assistance from college advising directors or from the admissions office in completing this section.

The addition of this course will in no way restrict any curricular transfer articulation agreements, and may add options for students to transfer credits.

O. Current catalog copy: N/A

P. Proposed catalog copy: This course covers the fundamentals of programming in C. Topics include basic syntax, data types, declarations, expressions, variables, compiling, input/output, conditional statements, loops, arrays, functions, pointers, arrays, strings, structures, structures as parameters, array of
structures, reading/writing files, and the C preprocessor.

Department Curriculum Chair approver: Jason Johnson

Department Curriculum Chair comment:

Date: 08-OCT-2019

Department approver: Steve Carr

Chair comment:

Date: 08-OCT-2019
CS 1210: Introduction to C
Master Syllabus

**Basic Syntax and Types (1 week)**
C File Syntax and main()
C Basic Data Types

**Declarations and Expressions (2 weeks)**
Global vs. Local Declarations
Expressions and Assignment
Basic I/O
Basic Compilation

**Control Flow (2 weeks)**
Conditional Statements
Loops
Arrays

**Subroutines (3 weeks)**
Functions and Stack Layout
Parameters and Call-by-value Parameters

**Pointers (3 weeks)**
Pointers to Basic Types
malloc() and free()
Strings
Pointers and Call-by-reference Parameters

**Structures (2 weeks)**
Structures
Structures as Parameters
Array of Structures

**File I/O and Preprocessing (1 week)**
Reading and Writing a File
C Preprocessor

**Course Prerequisites**

CS 1120
Catalog Description

This course covers the fundamentals of programming in C. Topics include basic syntax, data types, declarations, expressions, variables, compiling, input/output, conditional statements, loops, arrays, functions, pointers, strings, structures, structures as parameters, array of structures, reading/writing files, and the C preprocessor.

Learning Outcomes

Students will be able to:

- Design, implement, test, debug, and document programs in C.
- Write simple and intermediate-level C programs that compile and run correctly.
- Demonstrate how to write and use functions, how the stack is used to implement function calls, and parameter passing options.
- Demonstrate use of use common data structures typically found in C programs such as arrays and structs.
- Write programs that perform explicit memory management.
- Use pointers effectively; understand the risks of memory leaks and strategies for minimizing such risks.