

I. CATALOG INFORMATION

- A. Discipline: COMPUTER SCIENCE (CS)
- B. Subject Code and Number: CS M10B
- C. Course Title: Object-Oriented Programming Using C++
- D. Credit Course units:
- Units: 4
- Lecture Hours per week: 3
- Lab Hours per week : 3
- Variable Units : No

- E. Student Learning Hours:
- Lecture Hours:
- Classroom hours: 52.5 - 52.5
- Laboratory/Activity Hours:
- Laboratory/Activity Hours 52.5 - 52.5
- Total Combined Hours** in a 17.5 week term: 105 - 105

- F. Non-Credit Course hours per week _____

- G. May be taken a total of: ☒ 1 ☐ 2 ☐ 3 ☐ 4 time(s) for credit

- H. Is the course co-designated (same as) another course: No ☒ Yes ☐
- If YES, designate course Subject Code & Number: _____

- I. Course Description:
- Provides an in-depth understanding of the fundamentals of object-oriented programming methodology. Covers encapsulation, classes, data and implementation hiding, message passing, inheritance, polymorphism, function overloading, operator overloading, parameterized functions, parameterized classes, virtual functions, exception handling, and an introduction to the Standard Template Library.

- J. Entrance Skills
- *Prerequisite: No ☐ Yes ☒ Course(s)
CS M10A
- *Corequisite: No ☒ Yes ☐ Course(s)

- Limitation on Enrollment: No ☒ Yes ☐

- Recommended Preparation: No ☒ Yes ☐ Course(s)

- Other: No ☒ Yes ☐

K. Other Catalog Information:

II. COURSE OBJECTIVES

Upon successful completion of the course, a student will be able to:

		Methods of evaluation will be consistent with, but not limited by, the following types or examples.
1	describe and apply standard object-oriented programming features such as encapsulation, classes, constructors and destructors, inheritance, polymorphism, and member access specifications.	Graded assignments Programming projects Quizzes Exams
2	describe and use C++ features such as virtual functions, operator overloading, pointers, and exception handling.	Graded assignments Programming projects Quizzes Exams
3	describe and design algorithms using object-oriented techniques to solve programming problems as well as analyze and apply those algorithms.	Graded assignments Programming projects Quizzes Exams
4	analyze software problems and develop solutions using testing and debugging techniques.	Graded assignments Programming projects Quizzes Exams
5	design and write programs using object-oriented concepts and proper program style techniques.	Graded assignments Programming projects Quizzes Exams
6	use separate compilation techniques in an integrated development environment, such as Visual Studio .NET, to manage programming projects.	Graded assignments Programming projects Quizzes Exams
7	describe and use the Standard Template Library (STL).	Graded assignments Programming projects Quizzes Exams

III. COURSE CONTENT

Estimated %	Topic	Learning Outcomes
Lecture (must total 100%)		
5.00%	Object-Oriented Programming Principles • Encapsulation • Data abstraction • Classes and objects • Data hiding • Client/Server approach	1, 3, 4, 5, 6
15.00%	Inheritance • Is-a relationship versus has-a relationship • Protected access • Fragile base classes • Base class initializer list • Instantiation order of base and derived classes	1, 2, 3, 4, 5, 6, 7
20.00%	Polymorphism • Function overloading • Operator overloading • Virtual functions	1, 2, 3, 4, 5, 6, 7
10.00%	Input and Output • Concept of C++ streams • Use of C++ standard input/output classes	1, 2, 3, 4, 5, 6, 7
5.00%	File Processing	1, 2, 3, 4, 5, 6, 7
10.00%	Parameterized Classes and Functions • Use of templates	1, 2, 3, 4, 5, 6, 7
5.00%	Exception Handling	1, 3, 4, 5, 6, 7
5.00%	Introduction to the Standard Template Library • Sequential and associative containers • Algorithms • Function pointers	1, 2, 3, 4, 5, 6, 7
25.00%	C++ Classes • Objects • Access specifiers • Data members (including pointers using dynamic memory) and member functions in a class • Pointers to objects, including multiple levels of indirection • Constructors and destructors • Class requirements when using dynamic memory in a class including the necessity of having a constructor, copy constructor, destructor, and assignment operator overload. • Initialization list • Use of const • Accessing class members • Friends • Has-a relationship (composition) • Static class members	1, 3, 4, 5, 6
	An instructor can vary the order of the topics, or add enrichment topics, without a loss of continuity in the Computer Science program	
Lab (must total 100%)		

20.00%	Project(s) to create a program or programs "from scratch" which will incorporate: • Creating classes and objects • Separate compilation with each class divided into a header file and an implementation file • Using private and public access specifiers • Using accessor and mutator functions • Correctly using objects • Accessing class members • Understanding which entities belong in the client and which belong in class(es)	1, 3, 4, 5, 6
20.00%	Project(s) to create a program or programs "from scratch" which will incorporate: • Operator overloading • Class pointers as well as class members which are pointers • Dynamic memory • Appropriate use of and and const in parameter lists	1, 2, 3, 4, 5, 6
20.00%	Project(s) to create a program or programs "from scratch" which will incorporate: • Inheritance • Proper division of tasks between base and derived classes • Virtual functions • Abstract base classes • Operator overloading • Function overloading • Dynamic arrays	1, 2, 3, 4, 5, 6
20.00%	Project(s) to create a program or programs "from scratch" which will incorporate: • Templates • Exceptions • Pointers and dynamic memory • Vectors • Input from a file	1, 2, 3, 4, 5, 6, 7
20.00%	Project(s) to create a program or programs "from scratch" which will incorporate: • File input and output including using and clearing stream error states • Using vectors • Using an Standard Template Library algorithm such as find_if • Using function pointers	1, 2, 3, 4, 5, 6, 7

IV. TYPICAL ASSIGNMENTS

A. Writing assignments

Writing assignments are required. Possible assignments may include, but are not limited to:	
1	answer homework questions, such as identify errors in code and explain the needed correction.
2	comment on computer programs, such as identifying efficient code vs. inefficient code.

B. Appropriate outside assignments

Appropriate outside assignments are required. Possible assignments may include, but are not limited to:	
1	research and comment on current computer/technology industry and social trends.
2	read professional publications in addition to the text assignments.
3	research and apply standard object-oriented programming features such as encapsulation, classes, constructors and destructors, inheritance, polymorphism, and member access specifications.

C. Critical thinking assignments

Critical thinking assignments are required. Possible assignments may include, but are not limited to:	
1	develop algorithms for programming assignments.
2	evaluate incorrect programs and/or program fragments to determine what errors occur and to correct the syntax and/or semantics of the problems identified.
3	apply C++ features such as virtual functions, operator overloading, pointers, and exception handling.

V. METHODS OF INSTRUCTION

Methods of instruction may include, but are not limited to:

- ☒ Distance Education – When any portion of class contact hours is replaced by distance education delivery mode (Complete DE Addendum, Section XV)
- ☒ Lecture/Discussion
- ☒ Laboratory/Activity
- ☒ Other (Specify)

Practice problems to develop proper programming skills and techniques

Student/instructor interaction using questions and answers

Projects and/or group work to enhance student understanding of the concepts
- ☐ Optional Field Trips
- ☐ Required Field Trips

VI. METHODS OF EVALUATION

Methods of evaluation may include, but are not limited to:

- ☐ Essay Exam

☒ Classroom Discussion

☒ Skill Demonstration
- ☒ Problem Solving Exam

☐ Reports/Papers/Journals

☐ Participation
- ☒ Objective Exams

☒ Projects

☒ Other (specify)

Written homework exercises to demonstrate knowledge of concepts and techniques

Programming assignments requiring students to analyze a problem, determine a solution, implement the solution using a programming language, and test and verify the program

Quizzes where students demonstrate their knowledge of the material

VII. REPRESENTATIVE TEXTS AND OTHER COURSE MATERIALS

Deitel, Paul, and Harvey Deitel. C++ How to Program. 10th ed. Pearson, 2016.

Gaddis, Tony. Starting Out With C++: From Control Structures Through Objects. 9th ed. Pearson, 2017.

Microsoft Visual Studio 2015 Express C++. Microsoft, 2015 ed.

Integrated Development Environment (IDE) for designing, writing, running, and testing C++ programs. This application is currently available as a free download from <http://www.microsoft.com/express/Downloads/>

VIII. STUDENT MATERIALS FEES

☒ No ☐ Yes

IX. PARALLEL COURSES

College	Course Number	Course Title	Units
California Polytechnic State University, San Luis Obispo	CPE/CSC 102	Fundamentals of Computer Science II	4
UCLA	COMPTNG 10A	Introduction to Programming (C++)	5
CSU Long Beach	CECS 174	Introduction to Programming and Problem Solving	3
CSU Sacramento	CSC 15	Programming Concepts and Methodology I	3
UC Santa Barbara	CMPSC 24	Problem Solving With Computers II	4
UC Davis	ENG CS 40	Introduction To Software Development And Object-Oriented: C++	4

X. MINIMUM QUALIFICATIONS

Courses Requiring a Masters Degree:

Master's in computer science or computer engineering OR Bachelor's in either of the above AND Master's in mathematics, cybernetics, business administration, accounting or engineering OR Bachelor's in engineering AND Master's in cybernetics, engineering mathematics, or business administration OR Bachelor's in mathematics AND Master's in cybernetics, engineering mathematics, or business administration OR Bachelor's degree in any of the above AND a Master's degree in information science, computer information systems, or information systems OR the equivalent.

XI. ARTICULATION INFORMATION

A. Title V Course Classification:

1. This course is designed to be taken either:

- ☐ Pass/No Pass only (no letter grade possible); or
☒ Letter grade (P/NP possible at student option)

2. Degree status:
Either ☒ Associate Degree Applicable; or ☐ Non-associate Degree Applicable

B. Moorpark College General Education:

1. Do you recommend this course for inclusion on the Associate Degree General Education list?
Yes: ☐ No: ☒ If YES, what section(s)?
- ☐ A1 - Natural Sciences - Biological Science
 - ☐ A2 - Natural Sciences - Physical Science
 - ☐ B1 - Social and Behavioral Sciences - American History/Institutions
 - ☐ B2 - Social and Behavioral Sciences - Other Social Behavioral Science
 - ☐ C1 - Humanities - Fine or Performing Arts
 - ☐ C2 - Humanities - Other Humanities
 - ☐ D1 - Language and Rationality - English Composition
 - ☐ D2 - Language and Rationality - Communication and Analytical Thinking
 - ☐ E1 - Health/Physical Education
 - ☐ E2 - PE or Dance
 - ☐ F - Ethnic/Gender Studies

C. California State University(CSU) Articulation:

1. Do you recommend this course for transfer credit to CSU? Yes: ☒ No: ☐
2. If YES do you recommend this course for inclusion on the CSU General Education list?
Yes: ☐ No: ☒ If YES, which area(s)?
- | | | | | | | | | | | | | | |
|----|--------------------------|----|--------------------------|----|--------------------------|----|--------------------------|-----|--------------------------|----|--------------------------|----|--------------------------|
| A1 | ☐ | A2 | ☐ | A3 | ☐ | B1 | ☐ | B2 | ☐ | B3 | ☐ | B4 | ☐ |
| C1 | ☐ | C2 | ☐ | D1 | ☐ | D2 | ☐ | D3 | ☐ | D4 | ☐ | D5 | ☐ |
| D6 | ☐ | D7 | ☐ | D8 | ☐ | D9 | ☐ | D10 | ☐ | E | ☐ | | |

D. University of California (UC) Articulation:

1. Do you recommend this course for transfer to the UC? Yes: ☒ No: ☐
2. If YES do you recommend this course for the Intersegmental General Education Transfer Curriculum (IGETC)? Yes: ☐ No: ☒

IGETC Area 1: English Communication

- ☐ English Composition
- ☐ Critical Thinking-English Composition
- ☐ Oral Communication

IGETC Area 2: Mathematical Concepts and Quantitative Reasoning

☐

☐ Mathematical Concepts

IGETC Area 3: Arts and Humanities

☐ Arts

☐ Humanities

IGETC Area 4: Social and Behavioral Sciences

☐ Anthropology and Archaeology

☐ Economics

☐ Ethnic Studies

☐ Gender Studies

☐ Geography

☐ History

☐ Interdisciplinary, Social & Behavioral Sciences

☐ Political Science, Government & Legal Institutions

☐ Psychology

☐ Sociology & Criminology

IGETC Area 5: Physical and Biological Sciences (mark all that apply)

☐ Physical Science Lab or Physical Science Lab only (non-sequence)

☐ Physical Science Lecture only (non-sequence)

☐ Biological Science

☐ Physical Science Courses

☐ Physical Science Lab or Biological Science Lab Only (non-sequence)

☐ Biological Science Courses

☐ Biological Science Lab course

☐ First Science course in a Special sequence

☐ Second Science course in a Special Sequence

☐ Laboratory Activity

☐ Physical Sciences

IGETC Area 6: Language other than English

☐ Languages other than English (UC Requirement Only)

☐ U.S. History, Constitution, and American Ideals (CSU Requirement ONLY)

☐ U.S. History, Constitution, and American Ideals (CSU Requirement ONLY)

XII. REVIEW OF LIBRARY RESOURCES

A. What planned assignment(s) will require library resources and use?

The following assignments require library resources:

Research, using the Library's print and online resources, on social trends in the

computer industry such as the use of graphical user interface to improve usability of programs.

- B. Are the currently held library resources sufficient to support the course assignment?

YES: ☒ NO: ☐

If NO, please list additional library resources needed to support this course.

XIII. PREREQUISITE AND/OR COREQUISITE JUSTIFICATION

Requisite Justification for CS M10A

- ☒ A. Sequential course within a discipline.

1. describe the basic components, syntax, and semantics of the C++ programming language.
2. analyze programming problems and design algorithms to solve those problems.
3. describe and identify sequential, selection, and iteration control structures.
4. describe and apply the concepts of structured programming including function usage and parameter passing.
5. describe and apply composite data types such as arrays and structures.
6. describe and apply user defined data types such as enumerations and structured data.
7. describe, analyze, and use the C++ string class.
8. describe and apply dynamic memory allocation using pointers.
9. describe and apply file input and output.
10. describe and identify good programming practice and style.
11. describe the basic organization of a computer system.
12. describe the basic organization of a computer system.
13. describe the basic components, syntax, and semantics of the C++ programming language.
14. analyze programming problems and design algorithms to solve those problems.
15. describe and identify sequential, selection, and iteration control structures.
16. describe and apply the concepts of structured programming including function usage and parameter passing.
17. describe and apply composite data types such as arrays and structures.

18. describe and apply user defined data types such as enumerations and structured data.

19. describe, analyze, and use the C++ string class.

20. describe and apply dynamic memory allocation using pointers.

21. describe and apply file input and output.

22. describe and identify good programming practice and style.

☐ B. Standard Prerequisite or Corequisite required by universities.

☐ C. Corequisite is linked to companion lecture course.

☐ D. Prerequisite or Corequisite is authorized by legal statute or regulation.
Code Section: _____

☐ E. Prerequisite or Corequisite is necessary to protect the students' health and safety.

☐ F. Computation or communication skill is needed.

☐ G. Performance courses: Audition, portfolio, tryouts, etc. needed.

XIV. WORKPLACE PREPARATION

CS M10B: Not Applicable

XV. DISTANCE LEARNING COURSE OUTLINE ADDENDUM

1. Mode of Delivery

☐ Online (course will be delivered 100% online)

☒ Online with onsite examinations (100% of the instruction will occur online, but examinations and an orientation will be scheduled onsite)

☒ Online/Hybrid (a percentage of instruction will be held online and the remaining percentage of instruction will be held onsite)

☐ Lab activities will be conducted onsite

☐ Televideo (Examinations and an orientation will be held onsite)

☐ Teleconference

☐ Other

2. Need/Justification

Improve general student access.

3. Describe how instructors teaching this course will ensure regular, effective contact with and among students.

The instructor will be available online for a minimum of 3 week-day hours/week, during the day at specific times. The instructor will communicate with students through the course management system which should include both synchronous tools (such as chat) and asynchronous tools (such as email and discussions).

Email is a tool primarily used for individual student contact. Students and the instructor can privately contact each other with questions, concerns, etc.

Discussions is a tool which is primarily used for contacting the entire class. Students can solicit help from other students, for example. Discussions can also be graded, encouraging students to participate in the class.

Use calendar and announcement tools to keep students informed of important events, deadlines, etc.

Collaborative learning would involve using software which allows students and the instructor to collaborate in real-time. These sessions can also be recorded so that students who were not able to participate can also benefit from these discussions. The instructor can talk with individual students or with student groups. Students can also collaborate with each other without the instructor.

4. Describe how instructors teaching this course will involve students in active learning.

Materials: These will be available online. Students will be able to download files and view them offline. Instructor can also provide links to websites.

Quizzes: These may be on a weekly or topic basis (or some other time frame as determined by the instructor) where students will test their knowledge of the material.

Assignments: These may include exercises through which students explore course concepts using the textbook and additional research. There may also be projects which require students to write programs. Students will submit their assignments online and get feedback from the instructor. This can be an iterative process in that students can receive feedback and then be able to improve their submittal, if necessary.

Email: Email is a tool primarily used for individual student contact. Students and the instructor can privately contact each other with questions, concerns, etc.

Discussions: Discussions is a tool which is primarily used for contacting the entire class. Students can solicit help from other students, for example. Discussions can also be graded encouraging students to participate in the class.

Collaborative learning: This would involve using software which allows students and the instructor to collaborate in real-time. These sessions can also be recorded so that students who were not able to participate can also benefit from these discussions. The instructor can talk with

individual students or with student groups. Students can also collaborate with each other without the instructor.

Grades: Students should be able to view their current grades online at any time.

- 5. Explain how instructors teaching this course will provide multiple methods of content representation.

The instructor can provide text, presentation slides, audio/visual material, programming examples, tutorials (which may be live or recorded), and links to websites.

- 6. Describe how instructors teaching this course will evaluate student performance.

Instructors will use standard techniques such as exercises, programs, quizzes, and exams to evaluate student performance. The online environment can allow the exercises and programs to be iterative as students can submit their work online and get feedback from the instructor. The instructor can communicate solutions to students by posting them online. Additionally, graded discussions can be used to provide another means of assessment.

XVI. GENERAL EDUCATION COURSE OUTLINE ADDENDUM

CS M10B: Not Applicable

XVII. STUDENT MATERIALS FEE ADDENDUM

CS M10B: Not Applicable

XVIII. REPEATABILITY JUSTIFICATION TITLE 5, SECTION 55041

CS M10B: Not Applicable

XIX. CURRICULUM APPROVAL

Course Information:

Discipline: COMPUTER SCIENCE (CS)

Discipline Code and Number: CS M10B

Course Revision Category: Outline Update

Course Proposed By:

Originating Faculty Esmaail Nikjeh 03/13/2017

Faculty Peer: _____

Curriculum Rep: Scarlet Relle 10/15/2017

Department Chair: _____

Division Dean: Mary Rees 03/13/2017

Approved By:

Curriculum Chair: Jerry Mansfield 11/10/2017

Executive Vice President: _____

Articulation Officer: Letrisha Mai 10/19/2017

Librarian: Mary LaBarge 10/19/2017

Implementation Term and Year: _____

Approval Dates:

Approved by Moorpark College Curriculum Committee: 11/07/2017

Approved by Board of Trustees (if applicable): _____

Approved by State (if applicable): 02/16/2018