

PALOMAR COLLEGE
COURSE OUTLINE OF RECORD FOR
DEGREE CREDIT COURSE

X Transfer Course X A.A. Degree applicable course
(check all that apply)

COURSE NUMBER AND TITLE: CSIS 220 Programming for Computer Science

UNIT VALUE: 4

MINIMUM NUMBER OF SEMESTER HOURS: 80

BASIC SKILLS REQUIREMENTS: Appropriate language and computational skills.

ENTRANCE REQUIREMENTS

PREREQUISITE: CSIS 160 Survey of Computer Science

COREQUISITE: None

RECOMMENDED PREPARATION: None

SCOPE OF COURSE:

Introduces object-oriented programming and design using Java. Focuses on implementation and testing of software in a platform-independent, event-driven, graphical user interface environment. Covers basic concepts of data representation, user interface design, and software engineering. *CSU; UC (CAN CSIS 12)*

SPECIFIC COURSE OBJECTIVES:

Successful students will be able to:

1. Follow the software development process (requirements analysis, design, implementation, test) in the development of small programs.
2. Employ good software engineering practices such as incremental development, encapsulation, data integrity checking, and adherence to style guidelines.
3. Construct simple object-oriented programs demonstrating use of interfaces and composition.
4. Create appropriate user interfaces for simple programs in a graphical user interface environment.
5. Select and model data appropriately using primitive and reference types.
6. Analyze and construct efficient and effective algorithms and translate to appropriate control structures in an implementation language.
7. Effectively use software development tools including libraries, compilers, editors, linkers and debuggers.

CONTENT IN TERMS OF SPECIFIC BODY OF KNOWLEDGE:

- I. Problem-Solving
 - A. Identification of objects, classes and interactions

- II. Object-Oriented Design
 - A. Abstraction and modeling
 - B. Classes and objects
 - C. Mutability
 - D. Design by contract
 - E. Pre- and post-conditions
 - F. Invariants
 - G. Encapsulation
 - H. Information hiding
 - I. Accessors and mutators
 - J. Properties
 - K. Specification
 - L. Unified Modeling Language

- III. Implementation
 - A. Coding style guidelines
 - B. Defensive programming
 - C. Efficiency

- IV. Errors
 - A. Types of errors
 - B. Debugging process

- V. Data Types
 - A. Primitive types (integers, floats, boolean, character)
 - B. Strings
 - C. Arrays of primitives

- VI. Expressions
 - A. Operators
 - B. Precedence
 - C. Values and side effects

- VII. Algorithms
 - A. Design
 - B. Searching
 - C. Sorting

- VIII. Virtual Machine Concepts
 - A. Bytecodes and platform independence
 - B. Architecture (loader, verifier, API, machine emulator)

- IX. Input/Output
 - A. File-based (text files only)
 - B. Interactive

- X. Program organization
 - A. Packages and import
 - B. Expressions and statements
 - C. Parameters (actual and formal)
 - D. Methods
 - E. Overloading

- XI. Program environment
 - A. Graphical user interface libraries
 - B. Console-based applications

- XII. Control structures
 - A. Sequence
 - B. Repetition (while, do-while, for)
 - C. Selection (if, if-else, switch)

- XIII. Classes
 - A. Interface
 - B. Reference types
 - C. Constructors
 - D. Instance and class variables and constants
 - E. Instance and class methods
 - F. Access modifiers (public, private)

- XIV. Error handling
 - A. Exceptions

- XV. Tools
 - A. Editors
 - B. Compilers and interpreters
 - C. Debuggers
 - D. Javadoc
 - E. HTML

REQUIRED READING:

Riley, David. The Object of Java – Introduction to Programming Using Software Engineering Principles – BlueJ Edition. Boston: Addison Wesley, 2003.

SUGGESTED READING:

None.

REQUIRED WRITING:

Problem solving exercises are assigned, requiring students to complete five or six computer programming labs. Each programming lab will consist of a hands-on exercise applying theory principles learned in class. Programs must be well documented (at least one paragraph) in terms of their overall design goals. Additionally, each subprogram must be documented (two or three sentences) as to its purpose and overall performance.

OUTSIDE ASSIGNMENTS:

Students are expected to spend a minimum of three hours per unit per week in class and on outside assignments, prorated for short-term classes.

There are written homework exercises within each section of each chapter which are assigned, requiring an average of one hour to complete. In addition, numerous programming assignments are assigned, each ranging from one to ten hours to complete by an average student.

INSTRUCTIONAL METHODOLOGY:

Check all that apply:

- lecture
- laboratory
- lecture-laboratory combination
- directed study

DISTANCE LEARNING:

This course may be offered as a distance learning course and meets Title 5 regulations 55370, 55372, 55374, 55376, 55378, and 55380.

Yes No

If yes, check all that apply:

- Television Course (Video one-way, e.g. ITV, video cassette, etc.)
- Online Course (Text one-way, e.g. newspaper, correspondence, electronic file, etc.)
- Two-Way Video Conferencing (Two-way interactive video and audio)
- One-Way Video Conferencing (One-way interactive video and two-way interactive audio)
- Computer Assisted Instruction (A specialized form of mediated instruction relying primarily on student access to information and prepared lessons or teaching materials through a computer terminal, but not under immediate supervision of a qualified instructor.)

GRADING POLICY AND STANDARDS (include methods of determining whether the stated objectives have been met by students):

Programming Assignments	45%
Midterm	25%
Final	30%

IS COURSE REPEATABLE FOR REASON(S) OTHER THAN DEFICIENT GRADE?

Yes No Number of times course may be taken for credit: 3

If yes, identify specific provision of Title 5 Division 2 section(s), 55761-55763 and 58161 which qualifies course as repeatable:

CONTACT PERSON:

Richard Stegman

SIGNATURES ON FILE