

CS 17.11 Course Outline as of Spring 2010**CATALOG INFORMATION**

Dept and Nbr: CS 17.11 Title: JAVA PROGRAMMING

Full Title: Java Programming

Last Reviewed: 1/24/2022

Units		Course Hours per Week		Nbr of Weeks	Course Hours Total	
Maximum	3.00	Lecture Scheduled	3.00	17.5	Lecture Scheduled	52.50
Minimum	3.00	Lab Scheduled	0	4	Lab Scheduled	0
		Contact DHR	0		Contact DHR	0
		Contact Total	3.00		Contact Total	52.50
		Non-contact DHR	0		Non-contact DHR	0

Total Out of Class Hours: 105.00

Total Student Learning Hours: 157.50

Title 5 Category: AA Degree Applicable

Grading: Grade Only

Repeatability: 00 - Two Repeats if Grade was D, F, NC, or NP

Also Listed As:

Formerly: CIS 17

Catalog Description:

Intended for students with previous programming experience. Topics include: Object-oriented programming principles, Java language constructs, the JDK (Java Developer's Kit), class libraries, multi-threading, networking, GUI (Graphical User Interface) development, applets and applications.

Prerequisites/Corequisites:

Course Completion of CS 10

Recommended Preparation:

Eligibility for ENGL 100 or ESL 100

Limits on Enrollment:**Schedule of Classes Information:**

Description: Intended for students with previous programming experience. Topics include Object-oriented programming principles, Java language constructs, the JDK (Java Developer's Kit), class libraries, multi- threading, networking, GUI (Graphical User Interface) development, applets and applications. (Grade Only)

Prerequisites/Corequisites: Course Completion of CS 10

Recommended: Eligibility for ENGL 100 or ESL 100

Limits on Enrollment:

Transfer Credit: CSU;UC.

Repeatability: Two Repeats if Grade was D, F, NC, or NP

ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:

AS Degree:	Area	Effective:	Inactive:
CSU GE:	Transfer Area	Effective:	Inactive:
IGETC:	Transfer Area	Effective:	Inactive:
CSU Transfer:	Transferable	Effective:	Summer 1996 Inactive:
UC Transfer:	Transferable	Effective:	Spring 2000 Inactive:

CID:

Certificate/Major Applicable:

Certificate Applicable Course

COURSE CONTENT

Outcomes and Objectives:

Students will:

1. Evaluate the rationale of the Java language.
2. Contrast Java and C++.
3. Design programs using object-oriented methods.
4. Create software using an integrated development environment.
5. Integrate the Java class libraries with the construction of new classes.
6. Test the efficiencies of multithreaded applications.
7. Construct graphical user interfaces.
8. Compare local I/O (Input/Output) facilities with networking in Java.
9. Evaluate, compare and contrast four design patterns.

Topics and Scope:

1. Object-oriented programming principles
 - A. Encapsulation
 - B. Inheritance
 - C. Polymorphism
2. Comparison with C/C++
 - A. Global variables
 - B. Pointers
 - C. Memory allocation
 - D. Header files
 - E. Preprocessor
3. Java language constructs
 - A. Types
 - B. Operators
 - C. Flow control

- D. Classes
- E. Packages and interfaces
- 4. JDK class libraries
 - A. .lang
 - B. .io
 - C. .util
 - D. .net
 - E. .awt
 - F. .applet
- 5. Threads and synchronization
 - A. Thread priorities
 - B. Synchronization
 - C. Messaging
- 6. Networking
 - A. Sockets for clients
 - B. Sockets for servers
 - C. URL (Uniform Resource Locator) connections
- 7. GUI development
 - A. Components
 - B. Layout manager
 - C. Menu container
- 8. Applets
 - A. HTML (Hypertext markup language) interface
 - B. Parameters
 - C. Initialization
 - D. Graphics
- 9. Applications
 - A. Parameters
 - B. Initialization
 - C. I/O

Assignment:

1. Reading, approximately 30 pages per week.
2. Write a minimum of 4 programs using the Java programming language.
3. Test and debug programs.
4. Write program documentation.
5. Take objective examinations including a final exam.

Methods of Evaluation/Basis of Grade:

Writing: Assessment tools that demonstrate writing skills and/or require students to select, organize and explain ideas in writing.

Written program documentation.

Writing 10 - 20%

Problem Solving: Assessment tools, other than exams, that demonstrate competence in computational or non-computational problem solving skills.

Writing, testing and debugging programs using the Java programming language

Problem solving
20 - 60%

Skill Demonstrations: All skill-based and physical demonstrations used for assessment purposes including skill performance exams.

None

Skill Demonstrations
0 - 0%

Exams: All forms of formal testing, other than skill performance exams.

Multiple choice, true/false, matching items, completion, programming exercises

Exams
20 - 60%

Other: Includes any assessment tools that do not logically fit into the above categories.

None

Other Category
0 - 0%

Representative Textbooks and Materials:

Head First Java (2nd edition) Sierra and Bates, O'Reilly: 2007
Thinking in Java (4th edition) Eckel, Prentice Hall: 2006