Northern Marianas College
CURRICULUM ACTION REQUEST

Effective Semester / Session: Spring 2022

Type of Action:

_	New
X	Modification
_	Move to Inactive (Stop Out)
_	Cancellation

Course Alpha and Number: CS 227

Course Title: Introduction to Programming

Reason for initiating, revising, or canceling:
This course guide has been updated to reflect the change of the textbook in accordance with the change in programming language to JAVA.

Michael D Rodgers, Jr 23 Nov 2021

Proposer Date

Velma Deleon Guerrero 29 November 2021

Interim Director, School of Business Date

Adam Walsh 12.02.21

Language & Format Review Specialist Date

Ajani Burrell 11.29.2021

Academic Council Chair Date

Dr. Randy Yates

Dean, Academic Programs and Services Date

1. **Department**
   School of Business

2. **Purpose**
   In this introductory computer programming course, students will be taught the fundamentals of programming that are applicable to any programming language. All fundamental programming concepts will be taught in this course using Java, a computer programming language. Students obtaining the A.A.S. in Business Administration degree with the Computer Applications emphasis are the target population of this course.

3. **Description**

   A. Required/Recommended Textbook(s) and Related Materials
   Required:
   Lysecky, R & Lizarraga, A. (n.d.). Programming in Java with zyLabs. *zyBooks a Wiley Brand*. https://www.zybooks.com/catalog/programming-in-java/#toggle-id-1.

   Recommended: None

   B. Contact Hours
   1. **Lecture:** 3 per week / 45 per semester
   2. **Lab:** Classes held in computer classroom/lab
   3. **Other:** None

   C. Credits
   1. **Number:** 3
   2. **Type:** Regular Degree Credits

   D. Catalogue Course Description
   The course presents such fundamentals of programming as data types, operators, control structures, arrays, strings, and functions. It prepares students to develop applications using the Java programming language. Students will be able to build useful programs using Java constructs while learning the basics of structured and object-oriented programming language and techniques. They will also be introduced to GUI and Web-based programming in Java. The course includes advanced coverage of arrays, inheritance, and GUI concepts. Prerequisites: CS222, MA132. English Placement Level: EN101. Math Placement Level: MA 161 (Offered: Spring).

E. Degree or Certificate Requirements Met by Course
This course is a required course for the A.A.S. degree in Business Administration with an emphasis in Computer Applications, and several associated Certificates of Achievement in the computer field. It also serves as a computer elective course for other degree options.

F. Course Activities and Design
Course activities include lectures and demonstrations using a computer and screen projector. Assignments involve creating simple programs to demonstrate proficiency. Individual coaching is used to discover and eliminate syntax errors and other program bugs. Exams assess students' understanding of the materials learned. A final project assesses the students' ability to apply the programming concepts.

4. Course Prerequisite(s); Concurrent Course Enrollment
Prerequisites: CS222 and MA132
Concurrent Course Enrollment: None

Required English/Mathematics Proficiency Level(s)
English Placement Level: EN101
Mathematics Placement Level: MA132

5. Estimated Cost of Course; Instructional Resources Needed
Cost to the Student: Tuition for a 3-credit course; lab fees, storage device for backup: USB Flash Drive–2GB or higher and the cost of the textbook and/or eBook (Zybooks).

Cost to the College: Instructor’s salary and/or any additional costs to NMC.

Instructional resources needed for this course include: computer hardware and software, computer supplies, computer maintenance, lab aides, stipends and reproduction of teaching supplies, instructor’s computer console, computer projector and projection screen, sound card and speakers, whiteboard, whiteboard markers, photocopied handouts, and appropriate reference materials.

6. Method of Evaluation
Students will be evaluated by periodic quizzes and exams, group participation in problem solving exercises, and practical application project assignments.

7. Course Outline

This is a topical outline and does not necessarily indicate the sequence in which the material will be presented.

1.0 Introduction to Java

2.0 Variables/Assignments

3.0 Branches

4.0 Loops

5.0 Arrays

6.0 User-Defined Methods

7.0 Objects and Classes

8.0 Memory Management

9.0 Input/Output

10.0 Inheritance

11.0 Recursion

12.0 Exceptions

13.0 Generics

14.0 Collections

15.0 GUI

16.0 JavaFX

17.0 Seraring and Sorting Algorithms

8. Instructional Goals

The course will introduce students to:

1.0 Designing elementary computer algorithms;

2.0 Developing small Java programs that implement basic algorithmic design;

3.0 Organizing and documenting program code;

4.0 The differences between integer, floating point, and character string data;

5.0 Problem analysis and designing a solution algorithm that will correctly solve the problem;

6.0 Implementing the algorithm by coding it into a programming language;

7.0 Debugging programs by removing syntax and logic errors;

8.0 Declaring and initializing variables using self-documenting identifiers;

9.0 Function parameters, arguments, definitions, function calls, and return values;

10.0 Ability to Java data types with a Java program;

11.0 Java principles to accomplish data input, output, selection, and repetition structures. Use selection control structures and logical operations including IF, IF...ELSE, nested IF, SWITCH statements, AND, OR, NOT, WHILE, DO...WHILE, FOR and ARRAY;

12.0 Concepts of object-oriented programming including inheritance, class and interface; and

13.0 Designing and displaying simple Java applets using sounds, images, and graphics.

9. Student Learning Outcomes

Upon successful completion of this course, students will be able to:

1.0 Read, interpret, analyze, and explain introductory Java programs;

2.0 Use editors to compose programming code and compilers to produce executable software;

3.0 Organize program code into modules using methods following the principles of modularity and abstraction;

4.0 Write, document, test, and debug Java programs; making use of variables, expressions, selection, and looping statements;

5.0 Assemble data and methods into classes at an introductory level following principles of encapsulation and data hiding;

6.0 Make use of arrays to store and process lists of data;

7.0 Analyze problems and develop computer algorithms to solve novel problems;

8.0 Understand the difference between the following variable data types: integer, floating point, and character string, array;

9.0 Test and debug the program by removing syntax and logic errors;

10.0 Implement the algorithm by coding it into a programming language;

11.0 Declare and initialize variables using self-documenting identifiers;

12.0 Implement selection control structures using if/else, else/if, switch;

13.0 Implement repetition control structures using while, do/while, for;

14.0 Understand the concepts of inheritance, polymorphism, and information hiding;

15.0 Derive classes to access constructors, data fields, and methods of a superclass.

16.0 Use the exception-handling mechanisms built into the Java language, including the try, catch, finally, and throws keywords.

17.0 Discover how to work with input and output files;

10. Assessment Measures of Student Learning Outcomes

Assessment of student learning may include, but not be limited to, the following:

1.0 Class Attendance;

2.0 Classwork/Participation;

3.0 Homework Assignments;

4.0 Quizzes;

5.0 Practical Project Assignments; and

6.0 Periodic Exams.