American University of Madaba

Faculty of Information Technology
Department of Computer Science
Course Syllabus
Spring Semester 2022-2023

1. Course Specifics

Course Title	Programming Fundamentals Lab
Course Number	401120
Course Prerequisite(s)	None
Course Co-requisite(s)	None
Credit Hours	3 Cr. H
Lecture Days & Time	
Class Location	
Course Instructor/s	Dr. Mohammad Bany Taha , Eng. Ansam
Syllabus Approved by:	
b. Academic Committee	
a. Department Chair	Dr. Ahmad Ababneh

2. Academic Staff Specifics

Name	Academic Rank	Office Number, Tel. Ext.	Office Hours	E-mail Address
Dr. Mohammad Bany Taha	Assistant Professor	BA-G29,	Mon,Wed 2:30 pm – 4:00 pm	m.taha@aum.edu.jo
Eng. Ansam	Lab Supervisor	SA-G06,	Wed 10:00 am – 1:00 pm	a.wlaidat@aum.edu.jo

3. Course Components

Resources:

Students are expected to take notes during the lectures, in addition to making use of the following resources:

Textbook:
Title : Java For Dummies (For Dummies (Computer/Tech)) 8th Edition by Barry Burd.
Publisher for Dummies 2022.

Course References:

1. Java How to Program, Early Objects (12th Edition) Author: Harvey M. Deitel, Paul J. Deitel Publisher: *Prentice Hall*
2. Java in a Nutshell, ISBN:978-0-596-00773-7, David Flanagan, 5th Edition.
3. Java: An Introduction to Computer Science and Programming,

Links on Web Research, Supplemental Soft Sources:

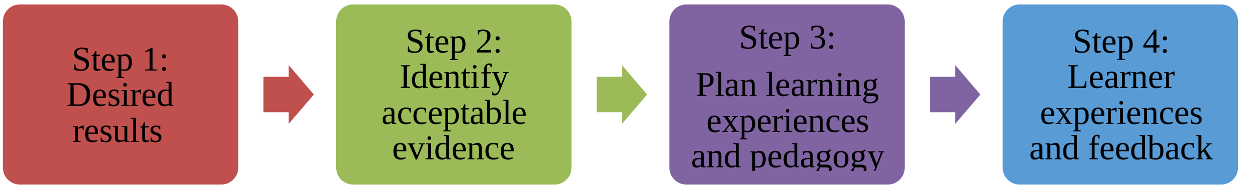
NetBeans 7	Java IDE	Free	http://netbeans.org/community/releases/70/
Eclipse 3.7	Java IDE	Free	http://www.eclipse.org/downloads/
Online Compiler		Free	Available online https://www.onlinegdb.com/online_java_compiler

			https://www.jdoodle.com/online-java-compiler/

Course Description:

This course introduces beginning computer programming skills with a guide to developing applications. The students can develop useful programs while learning the basics knowledge of programming skills. This course also provides the practical example that solved using programming languages. By learning the course topics, the students will be motivated to find real life problems in computer science and solve it using programming.

Online experiential learning based on Content Representation (CoRe) tool (Hume & Berry, 2010) and the Backward Design tool (Wiggins & McTighe, 2005).


Step 1: Desired results

Step 2: Identify acceptable evidence

Step 3: Plan learning experiences and pedagogy

Step 4: Learner experiences and feedback

Step One: Identify desired results from the experiential learning session

1. What do learners need to be able to do related to the topic/concept that will be covered in the lesson?
   - To understand general computer components
   - To understand the interaction between high level and low level languages.
   - To understand basic programming skills.
   - To understand the life-cycle of programming skills.
   - To solve mathematical and computation problem.

2. Why is it important for the learners to know this topic/concept?
   - To Demonstrate programming methodologies
   - To determine and evaluate programming technique that you learn.
   - To be able to select the appropriate methodology for broad wide of mathematics and computing problem.

3. What are the difficulties/limitations connected with teaching this concept online?
   - The missing of applying the theoretical material (through lab, self-thinking to solve problems) will increase the gap and make the students far from learning programming skills.
   - Missing classes will render more difficulty.

4. What do I know about the learners' learning which influences my online teaching of this concept?
   - The student enough knowledge about online tools.

- The students have enough knowledge about online classes management.

Step Two: Determine acceptable evidence: How will you know that the learners achieved the desired results?

2. How will I ascertain learner understanding of or confusion about the concept?
    - Discussion, the class will be active and every student expect to answer questions during the class.
    - HomeWorks, assignments, and Quizzes are part of the class.

3. What informal checks will I put in place to ascertain learner understanding?
    - Discussion.

4. What formal checks will I put in place?
    - Quizzes
    - HomeWorks
    - Midterm
    - Project
    - Final exam

Step Three: Plan Experiential Learning Experiences and Instruction

1. What knowledge and skills do learners need to perform effectively and achieve desired results?
    - Carefully follow up with synchronous class.
    - Carefully practices and apply the programming concepts in IDEs.

2. What experiential activities will be implemented to support learners with the knowledge and skills to be learned?
    - Write a joint group-specific reflection report on the watched video or lecture.
    - Crowd-pleasing activity racing across the online screen with correct answers.
    - Conduct a persuasive speaking platform for learners to discuss basic concepts in programming skills.
    - Write an individual reflection report commenting on the learner's experience of interaction, deliberation, and interpretation of basic programming principles.
3. What knowledge and skills will need to be taught and coached?
    a. Knowledge areas:

        i. Developing programming skills.

    b. Skills
        i. Cognitive and intellectual skills (discussion, brain storms for general problems)
        ii. Communication skills (personal and academic):
            a)
            b) Practical Skills: Use relevant skills learnt in this course for professional

and personal development.

c) Communicate fundamental system analysis/design concepts with formal means.

d) Demonstrate commitment to ethics, autonomy and professionalism in the workplace and others.

iii. Professional skills:

- Ability to work in teams
- Ability to use available tools to enhance software development process.

4. How should this be taught in an experiential online manner to achieve the goals?

- Using GoogleMeet to deliver asynchronous session.
- Using E-learning to deliver course documents (assignments, Self-reading, videos, …ect).
- Open-source software tools to solve project and mini-projects.
- using AnswerGarden for real time audience participation, online brainstorming, and classroom.

Step Four: Feedback: How will I assess learners' experiences with this online experiential session?

1. How will I receive feedback from the learners related to their experience?
   - Formative assessments
   - Summative assessments

2. Is the session coherent, learner-centric, and experiential?
   - Precisely, and will add that the session is also inclusive with an inclusionary paradigm set in place throughout the course.

Course Objectives:

1. Familiarize students with basic programming concepts in Java.
2. Provide students with the needed knowledge to use of primitive data types in Java.
3. Provide students with the needed knowledge regarding basic programming control structures for solving problems in java e.g. selection and iteration.
4. Train students to use methods technique in java and how to utilize them in modularizing programs and reduce programs complexity and enhance their understandability, readability, maintainability and code reusability.
5. Introduce basic and common data structures used in Java e.g. Arrays, Strings, etc and utilize them in order to solve common problems in programming.
6. Introduce students to the fundamental concepts of object-oriented programming techniques and apply them to solve common programming problems.
7. Extend the skills that helps student to move to advance levels in Java or other similar languages.

Course Learning Outcomes:

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

1. To understand the basics Java programming language. Be familiar with syntax and semantics, compilation and debugging techniques.
2. To develop the ability to write Java code with ease while utilizing data type, control structures, methods, and classes.
3. To apply good programming techniques, such as the modularization of programs and the use of naming conventions and comments.
4. To address typical programming issues and try to address unique issues.
5. To understand the concept of java programming language's object-Oriented.

Mapping CLO's with LOC's

	LOC's											
Course Learning Outcomes	A1	A2	A3	A4	A5	B1	B2	B3	C1	C2	D1	D2
1			1			1						
2	1											
3	1				1			1				
4											1	
5												1

"Program Outcomes (POs) are available at the academic department"

Mapping CLO's with LoC's of general education

	LOC's											
Course Outcomes	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	D1	D2
1	1	1										
2												1
3				1		1				1	1	
4							1	1	1			
5			1									1

Teaching Methods:

The structure of this course is elaborated throughout lectures, assignments, class, homework & projects.

Assessment Instruments:

Allocation of Marks	
Assessment Instruments	**Mark**
MidTerm Exam	**30%**
Participation /Quizzes	**15%**
Project	**15%**

Final Exam (written unseen exam)	**40%**
Total	**100%**

Course Academic Calendar

Weeks	Basic and support material to be covered	Synchronous Activity	Asynchronous Activity
Week 1:	Introduction to computers and Java		HW1
Week 2:	Elementary Programming	Assignment 1	Hw2
Week 3:	Selection	Assignment 2	Hw3
Week 4:	Loops	Assignment 3	Hw4
Week 5:	Loops, Methods I	Assignment 4	Hw5
Week 6:	Methods II	Assignment 5	Hw6
Week 7:	Single Dimensional Arrays	Assignment 6	Hw7
Week 8:	Single Dimensional Arrays	Assignment 7	Hw8
Week 9:	Midterm (21/ April / 2022) Multi-Dimensional Array	Assignment 8	Hw9
Week 10:	Multi-Dimensional Array	Assignment 9	Hw10
Week 11:	Object Oriented I	Assignment 10	Hw11
Week 12:	Object Oriented II	Assignment 11	Hw12
Week 13:	Project Discussion	Assignment 12	Hw13
			project
	Final Exam		

Course Policies

University regulations apply to this course regarding class attendance, punctuality, exams, late submissions, absence with permission, penalties for cheating, and policies for assignments and projects, if any. Students should be aware of all those in addition to other rules and regulations stated and descried in the student handbook.