

**ATTACHMENT 2 (g)**

**Course Report**

**Kingdom of Saudi Arabia**

**The National Commission for Academic Accreditation & Assessment**

**Course REPORT  
(CR)**

**Programming 2  
CSI 221-Z**

**Dr. Wael Khedr**

A separate Course Report (CR) should be submitted for every course and for each section or campus location where the course is taught, even if the course is taught by the same person. Each CR is to be completed by the course instructor at the end of each course and given to the program coordinator

A combined, comprehensive CR should be prepared by the course coordinator and the separate location reports are to be attached.

## Course Report

For guidance on the completion of this template refer to the NCAAA handbooks or the NCAAA Accreditation System help buttons.

Institution	Majmaah University	Date of Course Report	21/3/1436
College/ Department	College of Science / Department of Computer science and Information		

### A. Course Identification and General Information

1. Course title	Programming2	Code #	CSI 221-Z	Section #	138	
2. Name of course instructor	Dr. Wael Khedr	Location	College of Science at Al-zulfi			
3. Year and semester to which this report applies:	First Semester 1435/1436 H					
4. Number of students starting the course?	6	Students completing the course?	6			
5. Course components (actual total contact hours and credits per semester):						
	Lecture	Tutorial	Laboratory	Practical	Other:	Total
Contact Hours	30	---	30	---	---	60
Credit	30	---	15	---	---	45

### B. - Course Delivery

1. Coverage of Planned Program			
Topics Covered	Planned Contact Hours	Actual Contact Hours	Reason for Variations if there is a difference of more than 25% of the hours planned
1. A review of control structures and data types with emphasis on structured data types and array processing, review syntax of functions and primitive data types.	8	16	Because the background of programming 1 was very weak for almost students .
2. Introduction to input / output file streams.	4	4	--
3. Array of pointers	4	8	Because the background of programming 1 was very weak for almost students .
4. Introduce to the object-oriented programming paradigm, focusing on the	8	12	Because writing programs with language C++ was very weak for

definition and use of classes along with the fundamentals of object-oriented design			almost students .
5. Class and method (constructor, overloading , method)	8	8	---
6. Pointers and Iterators	8	8	---
7. Class Inheritance	4	0	There was no time to teach class inheritance ,polymorphism and exception because the pointers is finished in last week of term.
8. Polymorphism	4	0	
9. Exception Handling	8	0	

2. Consequences of Non Coverage of Topics For any topics where the topic was not taught or practically delivered, comment on how significant you believe the lack of coverage is for the course learning outcomes or for later courses in the program. Suggest possible compensating action.		
Topics (if any) not Fully Covered	Effectuated Learning Outcomes	Possible Compensating Action
Polymorphism	Students can not using a hierarchy of classes that are related by inheritance.	It can be taught in next level through Data Structure course.

### 3. Course learning outcome assessment.

	List course learning outcomes	List methods of assessment	Summary analysis of assessment results
1	Students will have a skills for upgrade their simple programs in Dev C++.	Written Exam Homework assignments Lab assignments Class Activities Quizzes	
2	Students will have an understanding of programming based on object , and complex programming.	Written Exam Homework assignments Lab assignments Class Activities Quizzes Observations	
3	Students will understand the concepts of and techniques used in C++ programming like classes, polymorphism.		
4	Apply C++ program structure and the VC++ object.		
5	Students will be able to analyze programming problems .		
6	Students will learn to think about life solutions by programming skills.	Written Exam Homework assignments Lab assignments	

		Class Activities Quizzes	
7	Work in a group and learn time management	Homework assignments Lab assignments Class Activities	
8	Learn how to search for information through library and internet..		
9	Present a short report in a written form and orally using appropriate scientific language.		

Summarize any actions you recommend for improving teaching strategies as a result of evaluations in table 3 above.

- Individual presentations
- Brainstorming
- Small group discussion
- Whole group

4. Effectiveness of Planned Teaching Strategies for Intended Learning Outcomes set out in the Course Specification. (Refer to planned teaching strategies in Course Specification and description of Domains of Learning Outcomes in the National Qualifications Framework)

List Teaching Methods set out in Course Specification	Were these Effective?		Difficulties Experienced (if any) in Using the Strategy and Suggested Action to Deal with Those Difficulties.
	No	Yes	
<ul style="list-style-type: none"> <li>• Lectures</li> <li>• Homework</li> <li>• conversation</li> </ul>		√	
<ul style="list-style-type: none"> <li>• Conversation between student.</li> <li>• Indirected questions.</li> <li>• Work group for some cases.</li> </ul>		√	
<ul style="list-style-type: none"> <li>• Making groups and distributed tasks.</li> <li>• Presentation skills.</li> <li>• Skill constructive Monetary and dialogue and discussion with others</li> <li>• The ability to clearly express an opinion, and accept the opinions of others</li> </ul>		√	
<ul style="list-style-type: none"> <li>• E-mail</li> <li>• Web sit</li> </ul>		√	

**Note:** In order to analyze the assessment of student achievement for each course learning outcome, student performance results can be measured and assessed using a KPI, a rubric, or some grading system that aligns student work, exam scores, or other demonstration of successful learning.

### C. Results

#### 1. Distribution of Grades

Letter Grade	Number of Students	Student Percentage	Explanation of Distribution of Grades
A	0		
B	1	16.667%	
C	1	16.667%	
D	1	16.667%	
F	3	50 %	
Denied Entry	0	0 %	
In Progress	0	0 %	
Incomplete	0	0 %	
Pass	3	50 %	
Fail	3	50 %	
Withdrawn	0	0 %	

#### 2. Analyze special factors (if any) affecting the results

#### 3. Variations from planned student assessment processes (if any) (see Course Specifications).

##### a. Variations (if any) from planned assessment schedule (see Course Specification)

Variation	Reason

b. Variations (if any) from planned assessment processes in Domains of Learning (see Course Specification)	
Variation	Reason

4. Student Grade Achievement Verification (eg. cross-check of grade validity by independent evaluator).	
Method(s) of Verification	Conclusion
Interview students, including answers and model answer sheet and learning resources for decision	Good result

#### D. Resources and Facilities

1. Difficulties in access to resources or facilities (if any)	2. Consequences of any difficulties experienced for student learning in the course.
---	---

#### E. Administrative Issues

1 Organizational or administrative difficulties encountered (if any)	2. Consequences of any difficulties experienced for student learning in the course.
--	---

#### F Course Evaluation

1 Student evaluation of the course (Attach survey results report The main factor of low success 's students is that their skills of programming is very bad Because the background of programming 1 was very weak for almost students
a. List the most important recommendations for improvement and strengths. <b>Strengths:</b> <ul style="list-style-type: none"> <li>- The course is strongly related to the simulation real problems.</li> <li>- The course encourages students to work as a team.</li> </ul>

<ul style="list-style-type: none"> <li>- The course prerequisites are appropriate for the course.</li> <li>- The textbook for this course and the level of the textbook are appropriate for this course.</li> </ul> <p><b>Recommendations for improvement:</b></p> <ul style="list-style-type: none"> <li>- Providing students with more practical programs in C++ .</li> <li>- Providing students with more concepts of programming that form a background for this course.</li> <li>- Encourage students to work as a team to implement real software projects.</li> <li>- Encourage students not to delay the beginning of the lecture.</li> </ul>
<p>b. Response of instructor or course team to this evaluation</p> <p>The course team acknowledges these recommendations for improvement.</p>
<p>2. Other Evaluation (e.g. by head of department, peer observations, accreditation review, other stakeholders)</p>
<p>a. List the most important recommendations for improvement and strengths</p>
<p>b. Response of instructor or course team to this evaluation</p>

### G. Planning for Improvement

1. Progress on actions proposed for improving the course in previous course reports (if any).			
Actions recommended from the most recent course report(s)	Actions Taken	Results	Analysis
a. Acquire students with more skills in concepts of programming.	<ul style="list-style-type: none"> <li>- More examples are added</li> <li>- An extra exercises and solved problems are added.</li> </ul>	Reasonable results	

b. Encourage students to use/apply the programming language in solving any real life problems	Make methods how can simulate the model problem by using programming languages concepts	Reasonable results	
c. Encourage students to attend extra hours through workshop in programming language .	Make a workshop in programming languages concepts for all students.	Reasonable results	
d. Encourage students not to attend lectures late	- Explain the importance of attending a full lecture - Give less important information at the beginning of each lecture	Reasonable results	

2. List what actions have been taken to improve the course (based on previous CR, surveys, independent opinion, or course evaluation).

- The use of programming to solve a real life problems.
- Enable students to prepare and make presentations.
- Increase related scientific activities.
- More examples are added.
- An extra exercises and solved problems are added.
- Explain the importance of attending a full lecture.
- Give less important information at the beginning of each lecture.

3. Action Plan for Improvement for Next Semester/Year

Actions Recommended	Intended Action Points and Process	Start Date	Completion Date	Person Responsible
a. Bridge the gap between up-to-date information and reference text books	- Give students the formal and theoretical bases in software programming as (Matlab). - Give students more implementation exercises that cover their understanding of the course.	1436	1437	Course coordinator



b. Overcome the problem of attending lectures late.	- Explain the importance of attending a full lecture. - Give less important information at the beginning of each lecture.	1436	1437	Course coordinator
c. Overcome the problem of insufficient background in programming concepts.	- Adding more examples and case studies. - Solving extra exercises. - Make a workshop in programming languages concepts for all students.	1436	1437	Course coordinator

Name of Course Instructor: **Dr. Wael Khedr**

Signature: \_\_\_\_\_

Date Report Completed: 21/ 03/ 1436

Program Coordinator: **Associate Prof. Yosry Azzam**

Signature: Yosry Azzam

Date Received: \_\_\_\_\_