

ATTACHMENT 2 (e)

Course Specifications

Kingdom of Saudi Arabia

The National Commission for Academic Accreditation & Assessment

**Course Specifications
(CS)**

Introduction to Programming With C++

CSC 115

1439H-2018G

Course Specifications

Institution	King Saud University	Date of Report	F.S. 1438-1439 H
College/Department	College of Science / Department of Mathematics		

A. Course Identification and General Information

1. Course title and code: CSC 115 (Introduction to Programming With C++)			
2. Credit hours 4 (3-0-2)			
3. Program(s) in which the course is offered. (If general elective available in many programs indicate this rather than list programs) Department of Mathematics			
4. Name of faculty member responsible for the course			
5. Level/year at which this course is offered 3rd Mathematics			
6. Pre-requisites for this course (if any) ICT 140			
7. Co-requisites for this course (if any) None			
8. Location if not on main campus			
9. Mode of Instruction (mark all that apply)			
a. Traditional classroom	<input checked="" type="checkbox"/>	What percentage?	<input type="text" value="100%"/>
b. Blended (traditional and online)	<input type="checkbox"/>	What percentage?	<input type="text"/>
c. e-learning	<input type="checkbox"/>	What percentage?	<input type="text"/>
d. Correspondence	<input type="checkbox"/>	What percentage?	<input type="text"/>
f. Other	<input type="checkbox"/>	What percentage?	<input type="text"/>
Comments:			

B Objectives

1. What is the main purpose for this course?
 - a. **The course aims to prepare the student with the basic and general concepts of programming language C++ so that student can use it later in his scientific and practical field.**
 - b. **The course aims to prepare students to understand the compiler way by the establishment of different algorithms to solve the various issues in various areas.**
The course aims to relate the programming language C++ with Mathematics, Statistics and Operations Research so that the programming language can be the way to solve many issues facing the student in the area of his specialization.
 - c. **The course aims to teach students cooperation through joint teamwork in integrated teams to resolve issues in different ways using programming language C++.**
 - d. **The course aims to give students the ability to follow programming language through self-learning and to follow the latest developments and issues related to them .**
 - e. **Students are meant to understand the OO programming concepts.**
2. Briefly describe any plans for developing and improving the course that are being implemented. (e.g. increased use of IT or web based reference material, changes in content as a result of new research in the field)
 - **Electronic materials and computer based programs have been utilized to support the lecture course material.**
 - **Presentation of some important sites on the Internet that serves the student to be familiar with the updated versions of the programming language**

C. Course Description (Note: General description in the form to be used for the Bulletin or handbook should be attached)

1. Topics to be Covered		
Topic	No of Weeks	Contact hours
Introduction to programming	1	5
Overall structure of a C++ program • Compiling, linking and running programs	1	5
Data types • Variables and constants • Operators: arithmetic, relational and Boolean • Expressions • input & output	2	10
Control structures (Decision) • If statement • If-else statement • Switch statement	1	5
Control structures (Iterations) • While loop • Do-while loop • For loop	2	10
Array • One dimension array • Two dimension array	2	10
Introduction to classes	1	5
Methods and message passing	3	15
Introduction to Inheritance and polymorphism	2	10

2. Course components (total contact hours and credits per semester):						
	Lecture	Tutorial	Laboratory	Practical	Other:	Total
Contact Hours	45	30				75
Credit	3	1				4

3. Additional private study/learning hours expected for students per week.	4
4 hours weekly for the homework and pre-laboratory reports assignments.	

4. Course Learning Outcomes in NQF Domains of Learning and Alignment with Assessment Methods and Teaching Strategy
--

Course Learning Outcomes, Assessment Methods, and Teaching Strategy work together and are aligned. They are joined together as one, coherent, unity that collectively articulate a consistent agreement between student learning, assessment, and teaching.

The *National Qualification Framework* provides five learning domains. Course learning outcomes are required. Normally a course has should not exceed eight learning outcomes which align with one or more of the five learning domains. Some courses have one or more program learning outcomes integrated into the course learning outcomes to demonstrate program learning outcome alignment. The program learning outcome matrix map identifies which program learning outcomes are incorporated into specific courses.

On the table below are the five NQF Learning Domains, numbered in the left column.

First, insert the suitable and measurable course learning outcomes required in the appropriate learning domains (see suggestions below the table). **Second**, insert supporting teaching strategies that fit and align with the assessment methods and intended learning outcomes. **Third**, insert appropriate assessment methods that accurately measure and evaluate the learning outcome. Each course learning outcomes, assessment method, and teaching strategy ought to reasonably fit and flow together as an integrated learning and teaching process. **Fourth**, if any program learning outcomes are included in the course learning outcomes, place the @ symbol next to it.

Every course is not required to include learning outcomes from each domain.

	NQF Learning Domains And Course Learning Outcomes	Course Teaching Strategies	Course Assessment Methods
1.0	Knowledge		
1.1	<ol style="list-style-type: none"> 1. The student will know the basics of programming languages C++, the use of input and output functions and how to write from simple programs -at the beginning- to the most complex programs. 2. The student will know how to formulate algorithms as a way to solve problems and apply it to examples and general and a variety of issues 3. The student knows the development of programming style, using all main topics in the language: such as <ul style="list-style-type: none"> • selection statements • Control statements • repetition • nested expressions • Functions • matrices • classes and objects 4. Using a programming language student can solve mathematical problems and statistical generation of random numbers with different statistical distributions to find the properties of these distributions to carry out the simulation. <p>Student's knowledge of Internet sites specialized in the languages of programming to communicate and participate with other programmers.</p>	<ul style="list-style-type: none"> • In-class lecturing where the previous knowledge is linked to the current and future topics • Homework assignments <p>Tutorial discussions</p>	<ul style="list-style-type: none"> • In class short MCQs quizzes • Homework assignments • Major and final exams

1.2			
2.0	Cognitive Skills		
2.1	Student's knowledge with structured programming and object oriented programming	<ul style="list-style-type: none"> • Homework assignments • Problem solving in the tutorial / recitation sessions 	<ul style="list-style-type: none"> • In class short MCQs quizzes • Major and final exams <p>Checking the problems solved in the homework assignments</p>
2.2			
3.0	Interpersonal Skills & Responsibility		
3.1	Projects to be designed with other students	<ul style="list-style-type: none"> • Conducting group experiments and writing group reports • Solving problems in groups during tutorial / recitation sessions 	<ul style="list-style-type: none"> • discussion about submitted project and the report related to it.
3.2			
4.0	Communication, Information Technology, Numerical		
4.1			
4.2			
5.0	Psychomotor		
5.1			
5.2			

Suggested Guidelines for Learning Outcome Verb, Assessment, and Teaching

NQF Learning Domains	Suggested Verbs
Knowledge	list, name, record, define, label, outline, state, describe, recall, memorize, reproduce, recognize, record, tell, write

Cognitive Skills	estimate, explain, summarize, write, compare, contrast, diagram, subdivide, differentiate, criticize, calculate, analyze, compose, develop, create, prepare, reconstruct, reorganize, summarize, explain, predict, justify, rate, evaluate, plan, design, measure, judge, justify, interpret, appraise
Interpersonal Skills & Responsibility	demonstrate, judge, choose, illustrate, modify, show, use, appraise, evaluate, justify, analyze, question, and write
Communication, Information Technology, Numerical	demonstrate, calculate, illustrate, interpret, research, question, operate, appraise, evaluate, assess, and criticize
Psychomotor	demonstrate, show, illustrate, perform, dramatize, employ, manipulate, operate, prepare, produce, draw, diagram, examine, construct, assemble, experiment, and reconstruct

Suggested **verbs not to use** when writing measurable and assessable learning outcomes are as follows:

Consider Maximize Continue Review Ensure Enlarge Understand
Maintain Reflect Examine Strengthen Explore Encourage Deepen

Some of these verbs can be used if tied to specific actions or quantification.

Suggested assessment methods and teaching strategies are:

According to research and best practices, multiple and continuous assessment methods are required to verify student learning. Current trends incorporate a wide range of rubric assessment tools; including web-based student performance systems that apply rubrics, benchmarks, KPIs, and analysis. Rubrics are especially helpful for qualitative evaluation. Differentiated assessment strategies include: exams, portfolios, long and short essays, log books, analytical reports, individual and group presentations, posters, journals, case studies, lab manuals, video analysis, group reports, lab reports, debates, speeches, learning logs, peer evaluations, self-evaluations, videos, graphs, dramatic performances, tables, demonstrations, graphic organizers, discussion forums, interviews, learning contracts, antidotal notes, artwork, KWL charts, and concept mapping.

Differentiated teaching strategies should be selected to align with the curriculum taught, the needs of students, and the intended learning outcomes. Teaching methods include: lecture, debate, small group work, whole group and small group discussion, research activities, lab demonstrations, projects, debates, role playing, case studies, guest speakers, memorization, humor, individual presentation, brainstorming, and a wide variety of hands-on student learning activities.

5. Schedule of Assessment Tasks for Students During the Semester

	Assessment task (e.g. essay, test, group project, examination, speech, oral presentation, etc.)	Week Due	Proportion of Total Assessment
1			
2			
3			
4			
5			
6			
7			
8			

D. Student Academic Counseling and Support

1. Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice. (include amount of time teaching staff are expected to be available each week)

E. Learning Resources

1. List Required Textbooks

Problem Solving with C++. Walter Savitch. Pearson education limited 2012, Eight edition.

2. List Essential References Materials (Journals, Reports, etc.)

3. List Recommended Textbooks and Reference Material (Journals, Reports, etc)

2) Author Name: H. M. Deitel & P. J. Deitel

Book title: How to Program C, Introducing C++, and Java,

Publisher : Pearson Education International , Year : Latest Edition

4. List Electronic Materials (eg. Web Sites, Social Media, Blackboard, etc.)

5. Other learning material such as computer-based programs/CD, professional standards or regulations and software.

- **Software compiler**

Multi media associated with the text book and the relevant websites.

F. Facilities Required

Indicate requirements for the course including size of classrooms and laboratories (i.e. number of seats in classrooms and laboratories, extent of computer access etc.)

1. Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)

Computer lab

2. Computing resources (AV, data show, Smart Board, software, etc.)

- **Computer lab**
- **Software compiler and interpreter**

3. Other resources (specify, e.g. if specific laboratory equipment is required, list requirements or attach list)

G Course Evaluation and Improvement Processes

1 Strategies for Obtaining Student Feedback on Effectiveness of Teaching

- **Course evaluation by student**
- Students- faculty meetings**

2 Other Strategies for Evaluation of Teaching by the Program/Department Instructor

- **Departmental council discussions**
- **Discussions within the group of faculty teaching the course**

3 Processes for Improvement of Teaching

- **Conducting workshops given by experts on the teaching and learning methodologies**
- **Periodical departmental revisions of its methods of teaching**
- **Monitoring of teaching activates by senior faculty members**

4. Processes for Verifying Standards of Student Achievement (e.g. check marking by an independent member teaching staff of a sample of student work, periodic exchange and remarking of tests or a sample of assignments with staff at another institution)

- **Providing samples of all kind of assessment in the departmental course portfolio of each course**
- **Assigning group of faculty members teaching the same course to grade same questions for various students. Faculty from other institutions are invited to review the accuracy of the grading policy**

5 Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement.

- **The course material and learning outcomes are periodically reviewed and the changes to be taken are approved in the departmental and higher councils.**
- **The head of department and faculty take the responsibility of implementing the proposed changes.**

Faculty or Teaching Staff: _____

Signature: _____ Date Report Completed: _____

Received by: _____ Dean/Department Head

Signature: _____ Date: _____