**Kingdom of Saudi Arabia**

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

**Computing Department, Community College Dammam**
**University of Dammam**

# Course Specifications
# (CS)

## Programming Fundamentals

CS110

# Programming Fundamentals

## Course Specifications

| | |
|---|---|
| **Institution:** University of Dammam | **Date of Report** |
| **College/Department:** Dammam-Community College / Computer Science Department | |

## A. Course Identification and General Information

1. Course title and code: Programming Fundamentals **(**CS110**)**
2. Credit hours: 3 (2 Theoretical + 2 Practical)
3. Program(s) in which the course is offered.
(If general elective available in many programs indicate this rather than list programs)
Information system and Information technology tracks
4. Name of faculty member responsible for the course:
5. Level/year at which this course is offered: 1st Level / Year 1
6. Pre-requisites for this course (if any): None
7. Co-requisites for this course (if any):
8. Location if not on main campus
9. Mode of Instruction (mark all that apply)

| | | | |
|---|---|---|---|
| a. Traditional classroom | √ | What percentage? | %70 |
| b. Blended (traditional and online) | | What percentage? | |
| c. e-learning | √ | What percentage? | %30 |
| d. Correspondence | | What percentage? | |
| f. Other | | What percentage? | |

Comments:

## B. Objectives

1. What is the main purpose for this course?

By the end of this course, the student should be able to:

1. Discuss the importance of algorithms in the problem-solving process and using pseudo-code.
2. Describe the phases of program translation from source code to executable code and the files produced by these phases.
3. Design, implement, test, and debug a program that uses fundamental programming constructs.
4. Explain the taxonomy of programming language characteristics and their effect upon the structure.
5. Explain some accepted models of programming language semantics. This should include models for both an imperative and a functional language.
6. Acquire a sound understanding of the theory and practice behind procedural, Object-oriented, functional and logic programming languages.

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).

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

This course will provide students with skills and concepts that are essential to programming practice independent of the underlying paradigm and programming language. The course will be accompanied by a practical part (lab) in which the students will learn popular C programming tools/applications and programming techniques.

1. Topics to be Covered

| List of Topics | No. of Weeks | Contact Hours |
|---|---|---|
| 1. An overview of algorithms and problem-solving (problem solving strategies, role of algorithms in the problem-solving process, etc.) | 3 | 6 T + 6 P |
| 2. Fundamental programming constructs (variables, types, expressions, simple I/O, conditional and iterative control structures, functions, recursion, pointers, etc.). | 4 | 8 T + 8 P |
| 3. The study of programming language features and programming paradigms. Control, run-time environments, and semantics are examples of procedural, functional, logical, and object oriented programming. | 4 | 8 T + 8 P |
| 4. In practice the programming language used is C. The syntax aspect of language and some pragmatic aspects must be studied in laboratory. | 4 | 8 T + 8 P |

| 2. Course components (total contact hours and credits per semester): | | | | | | |
|---|---|---|---|---|---|---|
| | **Lecture** | **Tutorial** | **Laboratory** | **Practical** | **Other:** | **Total** |
| Contact Hours | 30 | | | 30 | | 60 |
| Credit | 30 | | | 15 | | 45 |
| 3. Additional private study/learning hours expected for students per week. | | | | | | 4 |
| 4. Course Learning Outcomes in NQF Domains of Learning and Alignment with Assessment Methods and Teaching Strategy | | | | | | |

| | **NQF Learning Domains And Course Learning Outcomes** | **Course Teaching Strategies** | **Course Assessment Methods** |
|---|---|---|---|
| **1.0** | **Knowledge** | | |
| 1.1 | Identify the standard coding techniques and good programming practices; | Lectures, Class discussions, Demonstrations | Major Exams, Assignments, Exercises |
| 1.2 | Explain the programming structure, components and syntax of C programming language. | | |
| **2.0** | **Cognitive Skills** | | |
| 2.1 | Simulate a given C program written in C; | Lectures, Class discussions, Demonstratio | Seat work, Major Exams, Machine Problem, Lab exams, Lab record |
| 2.2 | Design a program based on the requirement specifications using C. | | |
| **3.0** | **Interpersonal Skills & Responsibility** | | |
| 3.1 | Choose the most appropriate programming structure or element to solve a problem | Lectures, Tutorials, Class discussions, presentations | Major Exam, Machine Problem, Lab record |
| 3.2 | Formulate a procedure to solve a programming problem. | | |
| **4.0** | **Communication, Information Technology, Numerical** | | |
| 4.1 | Communicate and present results/information effectively; | Question-and-answer Method | Student Presentation, Machine Problem, Lab Report, Major Exam |
| 4.2 | Work effectively as an individual and team. | | |
| **5.0** | **Psychomotor** | | |
| 5.1 | N/A | N/A | N/A |

5. Course Learning Outcomes Mapping Matrix

Identify on the table below the Course Outcomes and Relationship to PLOs

| Course Learning Outcomes | Program Learning Outcomes |
|---|---|
| **1. Knowledge** | |
| **1.1** | **1.1** |
| **1.2** | **1.2** |
| **2. Cognitive skills** | |
| **2.1** | **2.3** |
| **2.2** | **2.1 , 2.2** |
| **3. Interpersonal Skills and responsibility** | |
| **3.1** | **3.1, 3.2** |
| **3.2** | **3.3** |
| **4. Communication IT and Numeral Skills** | |
| **4.1** | **4.2, 4.3** |
| **4.2** | **4.1** |
| **5. Psychomotor Skills** | |
| **5.1** | **N/A** |

6. 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 | First quiz | 3 | %5 |
| 2 | Second quiz | 6 | %5 |
| 3 | Midterm | 8 | %20 |
| 4 | Third quiz | 10 | %5 |
| 5 | Lab | 13 | %20 |
| 6 | Attendance/Participation | All weeks | %5 |
| 7 | Final | 17 | %40 |

## D. Student Academic Counseling and Support

| |
|---|
| 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)<br><br>• Each group of students is assigned to a member of staff who will be available for help and academic guidance office hours at specific 2 hours on daily basis. |

## E. Learning Resources

| |
|---|
| 1. List Required Textbooks<br>• Maureen Sprankle and Jim Hubbard, "Problem Solving and Programming Concepts", 9th Edition. |
| 2. List Essential References Materials (Journals, Reports, etc.)<br>• Robert Sedgewick, "Algorithms in C: Fundamentals, data structures, sorting, searching, and graph algorithms", 3rd Edition, Addison-Wesley 2001, ISBN: 0201756080<br>• Robert W. Sebesta, "Concepts of Programming Languages", 9th Edition, 2009.<br>• Herbert Schildt, "C: the complete reference", McGraw-Hill 2000, ISBN:0-07-212124-6<br>• Robert Sedgewick and Kevin Wayne, "Algorithms", 4th Edition, 2011 |
| 3. List Recommended Textbooks and Reference Material (Journals, Reports, etc) |
| 4. List Electronic Materials (eg. Web Sites, Social Media, Blackboard, etc.)<br>• Blackboard and Social Media |
| 5. Other learning material such as computer-based programs/CD, professional standards or regulations and software.<br>• CDs accompanied with the text book, power point lectures and essential references |

## 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.)<br>Classrooms:<br>• Furnished with a large central table or multiple small tables that can be grouped into one central table<br>• Designed for up to 25 students<br>• Size the room allowing 1sq meter per seat<br>Laboratories:<br>25 PC's (one for each students) |

| |
|---|
| 2. Computing resources (AV, data show, Smart Board, software, etc.)<br>• Smart Board, projector, internet, and whiteboard**.** |
| 3. Other resources (specify, e.g. if specific laboratory equipment is required, list requirements or attach list)<br>• No |

## G. Course Evaluation and Improvement Processes

| |
|---|
| 1. Strategies for Obtaining Student Feedback on Effectiveness of Teaching:<br>• Student questionnaires to be assessed by independent body.<br>• Assessment of course teaching strategies by independent body. |
| 2. Other Strategies for Evaluation of Teaching by the Program/Department Instructor:<br>• Student questionnaires to be assessed by department. |
| 3. Processes for Improvement of Teaching:<br>• Attending workshop, reading books, and the searching for e-resources.<br>• Revision of course contents, course specifications, and strategies every 5 years. |
| 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)<br>• Check marking by an independent member of staff of a sample of student work.<br>• Periodic exchange and remarking of a sample of assignments with a member of staff in another institution |

| |
|---|
| 5. Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement.<br>• Reviewing student's feedback.<br>• Update text books.<br>• Consulting other top universities course specifications and contents. |