Proposed Curriculum
Course Description
For
Department of Computer Science
# Course Description

## Course Name
English Language I

## Course Code
ENGL 103

## Credit Hours
6

## Contact Hours

<table>
<thead>
<tr>
<th>R</th>
<th>L</th>
<th>W</th>
<th>G</th>
<th>Tot</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>18</td>
</tr>
</tbody>
</table>

## Track
- [ ] Core
- ✔ CS
- ✔ IS
- [ ] Major
- [ ] CNET
- [ ] Elective

## Level
1st Level

## Prerequisite

<table>
<thead>
<tr>
<th>Core</th>
<th>Major</th>
<th>Elective</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔ CS</td>
<td>✔ IS</td>
<td>✗ CNET</td>
</tr>
</tbody>
</table>

## Course Description:
English 001 is a multi-skill course taught on an intensive basis with work on specific skills allotted as follows: Reading: 5 hrs weekly, Listening: 5 hrs weekly, Writing: 5 hrs weekly and Grammar: 3 hrs weekly.

## Course Objectives:
1. To improve reading skills,
2. To build vocabulary,
3. To practice speech recognition,
4. To practice sentence construction,
5. To develop writing skills to the paragraph level,
6. To present and review grammatical structure, and
7. To practice structures orally.

## Grading

<table>
<thead>
<tr>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>40%</td>
<td></td>
<td>10%</td>
<td>10%</td>
</tr>
</tbody>
</table>

## Text Book:

## Reference Books:
1. Improving Aural Comprehension, ISBN 0472086650.
3. Introduction to Academic Writing, ISBN 020169509X.
Course Name: Mathematics  
Course Code: MATH 100  
Credit Hours: 3  
Contact Hours: Lec. 3  
Lab. 0  
Tot. 3  
Track:  
- Core  
- Major  
- Elective  
Level: 1st Level

Course Description:

Course Objectives:
Upon completing this course the student should be able to:
1. Identify and use the basic rules of logarithms and exponentials.
2. Identify and use some methods for solving equations and inequalities.
3. Identify and use the basic relations and identities of trigonometry.
4. Identify and use the basic properties of the triangle and the circle.
5. Identify and use the equation of the line and circle.
6. Develop familiarity with strategies for solving word problems and problems requiring reasoning for their solution.

Grading  
- Mid-term: 40%  
- Project:  
- Quizzes: 10%  
- Final: 40%  
- Lab.:  
- Participation: 10%  

Text Book:  
R.E. Larson and R.P. Hostetler “Precalculus”

Reference Books:  
Guntram Mueller, Ronald I. Brent:”Just-in-Time Algebra and Trigonometry for Students of Calculus”  
Jeffery Cole and E.W. Sowkowsk:” Algebra and Trigonometry with Analytic Geometry”
**Course Description:**

This course introduces the main concepts of computer science. It includes the basics of computing: hardware, Software, Connectivity, and users, the different types and features of computers. It presents also the data types and data Representation. A Simple Computer System architecture is presented so to emphasize on main components, secondary storage devices, types of memory, Hardware, software and people. The principal Peripheral Devices are also presented: Input, Output and storage, Data preparation, Factors affecting input, Input Devices, Output Devices, Secondary Storage devices, Communication between CPU and input/output devices. Software aspects are introduced like Problems-Solving and programming: Algorithm development, Flowcharts, Looping, Some programming Features, Pseudo code, Some structured programming concepts, Documentation, as well as Programming Languages : Machine language and assembly language, High-Level and Low-Level languages, assemblers, compilers and Interpreters. Finally, the course presents the computer and communication aspects, as well as different features of operating Systems.

**Course Objectives:**

1. To provide students with a deeper understanding of some specialty within computer science.
2. To understand basic computer concepts.
3. Introduction, basic computer organization.
4. To know about different programming language.
5. To know about algorithm and flowchart.
6. To know about different operating system.

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>20%</th>
<th>Project</th>
<th>Quizzes</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Final</td>
<td>40%</td>
<td>Lab</td>
<td>20%</td>
<td>Participation</td>
</tr>
</tbody>
</table>

**Text Book:**

Peter Norton's Introduction To Computers Fifth Edition Student Edition by Peter Norton  
Algorithmic: "The Spirit of Computing"

**Reference Books:**

1 Fundamental Concepts in Computer Science, edited by Erol Gelenbe (Imperial College London, UK) & Jean-Pierre Kahane (Université de Paris Sud - Orsay, France)
Course Description

Course Name: Islamic Culture 1
Course Code: ISLM 101

Credit Hours: 2
Contact Hours: Lec. 2, Lab. 2, Tot. 2

Track: Core
Level: 1st LEVEL
Prerequisite:

Course Description:

The course aims to provide an understanding of Islamic culture and its various aspects. It covers the history and development of Islamic culture, its relationship with other cultures, and its impact on modern society. The course also explores the role of Islamic culture in contemporary issues such as globalization and cultural diversity.

Course Objectives:

1. Develop an understanding of the fundamental concepts of Islamic culture and its evolution.
2. Analyze the role of Islamic culture in the development of modern society.
3. Discuss the impact of Islamic culture on contemporary issues such as globalization and cultural diversity.
4. Evaluate the contributions of Islamic culture to the development of modern society.
5. Understand the importance of Islamic culture in promoting social harmony and cultural diversity.

Grading:

- Mid-term: 40%
- Project: 10%
- Quizzes: 10%
- Participation: 10%
- Lab.: 40%
- Final: 40%

Text Book:

Textbook: "Themes in Islamic Culture" by [Author]

Reference Books:

- "The Islamic Culture (First Edition)" by [Author]
- "The Islamic Culture (Second Edition)" by [Author]
Course Name: English Language II

Course Code: ENGL 104

Credit Hours: 3

Contact Hours: Lec. 3, Lab. 3, Tot. 6

Track: [ ] Core [ ] Major [ ] Elective

CS [ ] IS [ ] CNET

Level: 2nd Level

Prerequisite: ENGL 101

Course Description:
English 002 develops the student’s ability in the areas of reading, writing and skills on aspects pertinent to scientific usage in general and, specially, regarding computer science when possible. In the area of reading skills the student continues work on primary reading skills (scanning, information retrieval, etc.) while increasing the range of basic, sub-technical and scientific vocabulary. This includes works on specially scientific prefixes, suffixes and derivations. In the area of writing, basic skill (mechanics, organization, etc.) are expanded. Special attention is given to writing comparisons, descriptions explanations and definitions, especially as pertinent to scientific usage. In the area of study skills, the student gains experience in the use of the dictionary and other references instruments, especially in the areas of science.

Course Objectives:
1. To develop primary reading skills,
2. To build general and scientific vocabulary,
3. To develop mechanics of writing,
4. To build written organizational skills, and
5. To practice the use of reference instruments.

Grading:
- [ ] Mid-term 40%
- [ ] Final 40%
- [ ] Quizzes 10%
- [ ] Project
- [ ] Lab.
- [ ] Participation 10%

Text Book:

Reference Books:
## Course Description

**Course Code**: ISLM 102  
**Course Name**: Islamic Culture 2

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>Contact Hours</th>
<th>Track</th>
<th>Level</th>
<th>Prerequisite</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>2nd Level</td>
<td></td>
</tr>
</tbody>
</table>

### Core Courses
- **CS**: Computing Science  
- **IS**: Information System
- **CNET**: Computer Engineering

### Course Description:

Aims: Understanding of the Islamic beliefs, values, and ethics. The aim of this course is to provide students with a comprehensive understanding of religious beliefs and ethics. Students will be introduced to the basic concepts of Islamic religion and ethics through the study of the Quran and the Hadith. The course will also cover the importance of Islam in today's world and its role in society.

### Course Objectives:

1. Understand the beliefs and ethics of Islam and their application in daily life.
2. Be able to apply the principles of Islam to various situations.
3. Be able to discuss the importance of Islamic values in contemporary society.
4. Be able to apply the principles of Islamic ethics in decision-making.
5. Be able to understand the role of Islam in world politics and international relations.

### Grading:

- **Mid-term**: 40%
- **Lab.**: 40%
- **Project**: 10%
- **Quizzes**: 10%
- **Participation**: 20%

### Text Book:

1. *العقائد الإسلامية*  
2. *شرح العقيدة الطحاوية*

### Reference Books:

1. *نظام الإسلام: العقيدة والعبادة لسفيان بن عروة*  
2. *شرح العقيدة الطحاوية*
<table>
<thead>
<tr>
<th>Course Name</th>
<th>Principles of Physics</th>
<th>Course Code</th>
<th>PHYS</th>
<th>201</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>4</td>
<td>Contact Hours</td>
<td>Lec.</td>
<td>Lab</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Track</td>
<td>□ Core</td>
<td></td>
<td>□ Major</td>
<td>□ Elective</td>
</tr>
<tr>
<td></td>
<td>□ CS</td>
<td></td>
<td>□ IS</td>
<td>□ CNET</td>
</tr>
<tr>
<td>Level</td>
<td>3rd Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prerequisite</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course Description:**
Charges and electrical forces. Charge quantization and conservation. Insulators, conductors and semiconductors. Electric field and principle of superposition for electric field. Electric dipoles and the effect of electric field on it. Electric potential due to charges and potential difference. Equipotential surfaces. Magnetic field and magnetic force. Introductory to modern physics and semiconductors. N-type and P-type crystals. Semiconductor carrier properties and action. Depletion layer and electric potential through it. Diode and types of diodes.

**Course Objectives:**
The student will be able to
1. Develop an understanding of electrical phenomena,
2. Develop an understanding of the magnetic phenomena,
3. Give the student the basic principles needed to study electronic courses, and give the student good background and experience in solving and dealing with problems.

**Grading**
- □ Mid-term 40%
- □ Project
- □ Quizzes 10%
- □ Final 40%
- □ Lab.
- □ Participation 10%

**Text Book:**

**Reference Books:**
Course Description

Department of Computer Science

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

Course Name: Programming - 1
Course Code: COMP 011

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Lec. 2, Lab. 2, Tot. 4</td>
</tr>
</tbody>
</table>

Track: □ Core  ❋ Major  □ Elective

CS  ❋ IS  ❋ CNET

Level: 2nd Level

Course Description:
This knowledge area consists of those skills and concepts that are essential to programming practice independent of the underlying paradigm and programming language. Specific topics covered include: an overview of algorithms and problem-solving (problem solving strategies, role of algorithms in the problem-solving process, etc), fundamental programming constructs (variables, types, expressions, simple I/O, conditional and iterative control structures, functions, recursion, pointers, etc.), fundamental and advanced data structures and related algorithms (arrays, records, strings and string processing, stacks, queues, trees and files, searching, sorting, etc). Students should acquire some understanding of data internal representation in order to write programs more efficiently. In practice the programming language used is ANSI-C, the syntax aspect of language and some pragmatic aspects such as: comparison of interpreters and compilers and language translation phases must be studied in laboratory.

Course Objectives:
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. Describe the internal representation of numeric and nonnumeric data.
5. Write programs that use fundamental and advanced data structures.

Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%</td>
<td></td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
<td>Lab. 20%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Text Book:

Reference Books:
### Course Description

**Department of Computer Science**

**Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA**

**Course Code**

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Course Code</th>
<th>COMP</th>
<th>114</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming of Statistics &amp; Probability</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Contact Hours**

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core Major Elective</td>
<td></td>
</tr>
<tr>
<td>Track</td>
<td>☑</td>
</tr>
<tr>
<td>Level</td>
<td>4th</td>
</tr>
<tr>
<td>Prerequisite</td>
<td></td>
</tr>
</tbody>
</table>

**Course Description:**

The course begins by revision of set theory. Introduction to the theory of probability is covered. Conditional probabilities, linear independence and Bay's theorem is covered. Discrete random variables and continuous random variables, density and cumulative distribution functions is also covered. The course ends by introducing the Correlation coefficients and Regression analysis.

**Course Objectives:**

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

1. Defines the main statistical fundamentals like Random Variable, Random Experiment, Event, Sample Space, Qualitative and Quantitative data and Frequency distribution table
2. Defines Mean, Variance and Standard deviation of discrete and continuous random Variable.
3. Define and determine the density functions and cumulative distribution functions for discrete and continuous random variables.

Estimate the correlation and regression coefficients

### Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑ Mid-term</td>
<td>20%</td>
<td>☑ Project</td>
<td>☑ Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>☑ Final</td>
<td>40%</td>
<td>☑ Lab.</td>
<td>20%</td>
<td>☑ Participation</td>
</tr>
</tbody>
</table>

### Text Book:

**Text Book**

1-A. Bery & M. Hendy "The Fundamentals of Probabilities and Statistics". King Saud University, 2000

### Reference Books:

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Islamic Culture 3</th>
<th>Course Code</th>
<th>ISLM</th>
<th>103</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>2</td>
<td>Contact Hours</td>
<td>Lec.</td>
<td>Lab.</td>
</tr>
<tr>
<td>Track</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Core</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3rd Level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prerequisite</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course Description:**

المقصود بعلم الاقتصاد: المذهبان الاقتصاديان المتضاربان في عالم اليوم.تعريف علم الاقتصاد الإسلامي، خصائص الاقتصاد الإسلامي، مصادر الاقتصاد الإسلامي، الملكية في النظام الاقتصادي الإسلامي، المشاكل الاقتصادية وعلاج الإسلام لها، المعاملات في الإسلام وممارستها نع النظم الاقتصادية المعاصرة. أركان الاقتصاد الإسلامي، المعاملات الدولية، التسامح، الحسية ودورها في تنظيم السوق والرقابة على الأسواق، بالإضافة إلى حفظ الجزء (28) من القرآن الكريم.

**Course Objectives:**

1. حدّد جوانب النظام الاقتصادي في الإسلام من خلال النظرية والتطبيق في الواقع وحولها.
2. بوابات النظام الاقتصادي في الإسلام والنظم الاقتصادية الوضعية موضحاً أوجه الشبه والاختلاف.
3. يعرق المعاني والمصطلحات الاقتصادية الأساسية بلغة خاصة مع التدريجية للامية.
4. يستنتج أهمية وضرورة تطبيق النظام الاقتصادي الإسلامي لدى المسلمين وغير المسلمين وتاثر تطبيق النظام الوضعية.

**Grading:**

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mid-term</td>
<td>40%</td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
</tr>
<tr>
<td>Project</td>
<td></td>
</tr>
<tr>
<td>Lab.</td>
<td></td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Participation</td>
<td>10%</td>
</tr>
</tbody>
</table>

**Text Book:**

1) الثقافة الإسلامية: المستوى الثالث. تأليف: محمد الهاموك وآخرون

**Reference Books:**

1) التفكير الاقتصادي في الإسلام. تأليف: خالد عبد الرحمن أحمد
2) النظام الاقتصادي القرآني. تأليف: محمد فريد منفي.
## Course Description

### Department of Computer Science

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Discrete Mathematics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Code</td>
<td>MATH 202</td>
</tr>
<tr>
<td>Credit Hours</td>
<td>3</td>
</tr>
<tr>
<td>Contact Hours</td>
<td>3 Lec. 0 Lab. 3 Tot.</td>
</tr>
<tr>
<td>Track</td>
<td>☑ Core ☑ Major ☑ CNET ☑ IS ☑ Elective</td>
</tr>
<tr>
<td>Level</td>
<td>4th Level</td>
</tr>
<tr>
<td>Prerequisite</td>
<td>MATH 102</td>
</tr>
</tbody>
</table>

### Course Description:


### Course Objectives:

1. Upon completing this course the student should be able to
2. Identify integer functions and some important relations on the integers
3. Know counting permutations and combinations
4. Know generating functions and their role in counting and the method for computing theses functions
5. Know graphs and paths and their applications
6. Know trees and their use in searching

### Grading:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>40%</td>
<td></td>
<td>10%</td>
<td>10%</td>
</tr>
</tbody>
</table>

### Text Book:

Donald Knuth, et. al.: "Concrete mathematics: A Foundation for computer science"

### Reference Books:

Course Description:

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Islamic Culture 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Code</td>
<td>ISLM 104</td>
</tr>
</tbody>
</table>

| Credit Hours | 2 |
| Contact Hours | Lec 2 | Lab 2 | Tot 2 |

<table>
<thead>
<tr>
<th>Track</th>
<th>Core</th>
<th>CS</th>
<th>Major</th>
<th>IS</th>
<th>Elective</th>
<th>CNET</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Level</th>
<th>4th Level</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Prerequisite</th>
<th>4th Level</th>
</tr>
</thead>
</table>

Course Description:

The course aims to understand the Islamic perspective of women and its relation to the family and society. It covers issues related to the family, women's rights, and the relationship between the family and society. It also deals with the theoretical and practical aspects of women's role in the family.

Course Objectives:

1. Understand the historical and social aspects of women's rights and their role in the family and society.
2. Investigate the relationship between the family and society, and the role of women in it.
3. Study the theoretical and practical aspects of women's role in the family.
4. Explore the relationship between women's rights and the family.
5. Analyze the theoretical and practical aspects of women's role in the family.

Grading:

- Mid-term: 40%
- Project: 10%
- Quizzes: 10%
- Lab: 40%
- Participation: 10%

Text Book:

1) Reference Books:

- The Islamic perspective on women's issues
- Women's role in the family

Reference Books:

1) The Islamic perspective on women's issues
2) Women's role in the family
Course Name: Programming II  
Course Code: COMP 112  
Credit Hours: 3  
Contact Hours: Lec. 2 | Lab. 2 | Tot. 4  
Track: ☒ Core  ☒ Major  ☐ Elective  
CS  ☒  IS  ☒ CNET  
Level: 3rd Level  
Prerequisite: COMP 011  

Course Description:
This course continues the coverage of the fundamental concepts of Object Oriented Programming started in Programming I (COMP011). It covers more advanced concepts and topics such as relationships between classes, inheritance, polymorphism, abstract classes, error handling, interfaces, generics and data structures such as linked lists, stacks and queues, graphical user interface.

Course Objectives:
2. relationships between classes, inheritance, polymorphism, abstract classes
3. generics and data structures such as linked lists, stacks and queues, graphical user interface

Grading  
☒ Mid-term  20%  ☒ Project  ☒ Quizzes  10%  
☒ Final  40%  ☒ Lab.  20%  ☒ Participation  10%  

Text Book:  

Reference Books:  
### Course Description

**Department of Computer Science**

**Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA**

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Computational Geometry</th>
<th>Course Code</th>
<th>COMP 151</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>Contact Hours</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Track</td>
<td>Core</td>
<td>Major</td>
<td>Elective</td>
</tr>
<tr>
<td></td>
<td>CS</td>
<td>IS</td>
<td>CNET</td>
</tr>
<tr>
<td>Level</td>
<td>4th Level</td>
<td>Prerequisite</td>
<td></td>
</tr>
</tbody>
</table>

#### Course Description:

The course begins by introducing the fundamentals of computational geometry as a new tool for modelling and designing algorithms, the application of the computational geometry in many different areas like geology and biology is also introduced. Graph theory is introduced in the next chapter including representing graphs, graph isomorphism, Euler graph, Hamilton path, shortest distance problems, adjacency matrix and distance matrix. The theory of trees is covered including binary search tree algorithm, decision tree and game tree. Tree traversal and spanning tree are also covered in the chapter.

#### Course Objectives:

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

1. Knowing the fundamentals of computational geometry as a new tool for modelling and designing algorithms.
2. Solving problems related to data structure
3. Designing effective algorithms for data structure problems

#### Text Book:


#### Reference Books:

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Calculus</th>
<th>Course Code</th>
<th>MATH 201</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
<td>Contact Hours</td>
<td>Lec.</td>
</tr>
<tr>
<td>Track</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Level</td>
<td>4th Level</td>
<td>Prerequisite</td>
<td>MATH 100</td>
</tr>
</tbody>
</table>

**Course Description:**

**Course Objectives:**
Upon completing this course the student should be able to:
- Solve problems involving computing limits
- Identify continuous functions and their properties
- Understand and use the definition of derivative, and compute derivatives using the rules of differentiation
- Apply the derivative in problems involving graphing and in problems involving maximization and minimization
- Understand the concept of integral and use the basic techniques of integration to compute integrals
- Apply integrals in solving some geometric problems
- Understand the relationship between the integral and derivative (the fundamental theorem of calculus)
- Understand the concept of sequence and compute limits of sequences
- Understand the concept of series and some of the tests for convergence
- Find the (formal) series expansion for rational functions

**Grading:**
- Mid-term: 40%
- Project: 60%
- Quizzes: 10%
- Final: 40%
- Lab.: 60%
- Participation: 10%

**Text Book:**

**Reference Books:**
Course Description

Introduction to Information System

Course Code
INFS 111

Credit Hours
3

Contact Hours
Lec. 2  Lab. 2  Tot. 4

Track
Core  ☑  Major  ☑  Elective  ☑

Level
3rd  Level

Course Description:
This course aims to introduce students to the basic concepts and topics related to Information Systems (IS). It covers topics such as: systems concepts; system components and relationships; cost/value and quality of information; competitive advantages of information; specification, design, and re-engineering of IS; application versus system software; package software solutions; procedural versus non-procedural programming languages; object oriented design; database features, functions, and architecture; networks and telecommunication systems and applications; characteristics of IS professionals and IS career path; information security, crime, and ethics. Practical exercises may include developing macros, designing and implementing user interface and reports; developing a solution using database software.

Course Objectives:
The course objectives aim:
1. To provide an introduction to the Organizational uses of information to improve overall quality.
2. To present hardware, software, and related information technology concepts.
3. To provide concepts and skills for the specification and design or the re-engineering of organizationally related systems of limited scope using information technology.
4. To show how information technology can be used to design, facilitate, and communicate organizational goals and objectives.
5. To explain the concepts of individual decision making, goal setting, trust worthiness, and empowerment.
6. To show career paths in information system. To present and discuss the professional and ethical responsibilities of the IS practitioner.

Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mid-term</td>
<td>40%</td>
<td>Project</td>
<td></td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
<td>Lab.</td>
<td></td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Text Book:

Reference Books:
Course Name | Algorithms & Data Structures 1
---|---
Course Code | COMP 221
Credit Hours | 3
Contact Hours | Lec. 2 Lab. 2 Tot. 4
Track | ☑ Core ☑ Major ☑ Elective
CS | ☑ IS | ☑ CNET
Level | 5th Level
Prerequisite | 5th Level

Course Description:
Abstract Data Types; Performance Measurement: Time & Space Complexity, Big-O Notation, Recursion, Basic Data Structures: Lists, Stacks, Queues, Circular Queue, Priority Queue, Linked List, Circular linked List, Priority Linked List, Sorting, Bubble Sort, Insertion Sort, Selection Sort etc, Searching Linear search, binary Search etc.

Course Objectives:
1. To understand Performance Measurement of Algorithm.
2. To understand Basic Data Structure.
3. To understand the use of Data Structures.
4. To understand Sorting.
5. To understand Searching.

Grading
- Mid-term: 20%
- Project: 20%
- Quizzes: 10%
- Lab.: 40%
- Participation: 20%
- Final: 40%

Text Book:
Larry Nyhoff, ADTs Data Structures and Problem Solving with C++, Prentice-Hall.

Reference Books:
Course Description:
This course aims to discuss the basic concepts and design of database. It covers topics such as: data model, levels of abstraction, data independence, and concurrency control. Focuses on how to design databases for given problems, and how to use database effectively, these including ER model, key and participation constraints, weak entities, class hierarchies, aggregation and conceptual DB design using the ER model. Relational model: creating and modifying relation using query language, enforcing integrity constrains, ER to relational and view. Schema refinement and normal forms: Functional dependencies, reasoning about functional dependencies, normal forms, decompositions and normalization. Relational Queries: Relation algebra and calculus and commercial query languages. Object database systems: User defined abstract data type, structured types, objects; object identity; and reference type, inheritance, and database design for an ORDBMS. Students will be trained on some software tools such as: Oracle, Sybase, DB2, and Informix.

Course Objectives:
1. Describe and discuss the concepts of database design.
2. Students learn how to design a conceptual data model and logical database model, convert the logical database designs to physical designs and develop the physical database.
3. Students learn how to evaluate a set of query using relational algebra and calculus.
4. Students learn how to evaluate a set of query using query language.
5. Discuss and explain database design for an ORDBMS.

Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%</td>
<td>10%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Text Book:</th>
</tr>
</thead>
</table>

Reference Books:

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
</table>
**Course Description:**

There are four classifications of models: discrete or continuous, probabilistic or deterministic, static or dynamic, and open loop or closed loop. The course objective is to produce students who are capable of modelling and simulating discrete, probabilistic, dynamic, and open loop system as well analyzing, verifying and validating the simulations results. The purpose of this course is to provide students with a theoretical base in discrete-event modelling and simulation for applying concepts related to computer networks and information system modelling (random numbers, Monte Carlo methods, Probabilistic modelling, Queuing theory models, Markov models and chains, arrival laws, service laws, birth-dead process, stochastic process, stationary process, stochastic analysis, networks analysis and routing algorithms, verification and validation of simulation models). Discrete production systems are studied (time flow mechanism, Petri nets). Students should complete a major project using simulation models and a standard simulation language. Students will be trained on some software tools such as: ARENA, QNAP, and PETRI NETS.

**Course Objectives:**

1. Discuss the fundamental concepts of computer modeling and simulation.
2. Be proficient in the use of computer simulation tools.
3. Describe several established techniques for prediction and estimation.
5. Design, code, test, and debug simulation programs.
6. Design and build a simulation model for computer area (communication and computer networks, management information system).

**Grading:**

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25%</td>
<td>15%</td>
<td>10%</td>
<td>5%</td>
</tr>
<tr>
<td>Final</td>
<td>45%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Text Book:**


**Reference Books:**

Course Description:
The purpose of this course is to provide students with fundamental knowledge of object oriented programming (OOP). It emphasizes good software engineering principles and developing programming skills. Specific topics covered include: fundamental concepts of object oriented (classes, methods, instantiation, communication by message, encapsulation, inheritance, overriding, dynamic dispatch, polymorphism, etc.), advanced techniques of OOP (exceptions, multithreaded programming, etc.) and some interesting packages (I/O, strings, etc.). As an OOP programmer, student will be able to translate solution problem into object oriented form, he should acquire some understanding of object oriented concepts and tools such as the Unified Modeling Language (UML), this will give student a firm foundation on which to build high-quality software systems. In practice the programming language used is JAVA, as an introduction to JAVA language; students should acquire some understanding of abstraction mechanisms, JAVA Virtual Machines (JVM) and the byte code notion.

Course Objectives:
1. Describe the importance and power of abstraction in the context of virtual machines and explain the benefits of intermediate languages in the compilation process.
2. Justify the philosophy of object-oriented design and the concepts of encapsulation, inheritance and polymorphism.
3. Explain how abstraction mechanisms support the creation of reusable software components.
4. Acquire basics of how translate solution problem into object oriented form.
5. Design and implement simple programs in an object-oriented programming language.
6. Design and implement program that use exceptions and multithreads.

Text Book:

Reference Books:
Course Description:

Register transfer and micro operations: register transfer language, bus and memory transfer, arithmetic, logic and shift micro operations. Basic computer organization and design: instruction codes, computer registers and common bus systems, computer instruction set, timing and control, instruction cycle, memory reference instructions, input-output and interrupt instructions, complete computer description, and design of basic computer. Hardwired and micro programmed control: hardwired control methods, hardwired control examples, control memory, address sequencing, micro program example, and design of control unit. Central processing unit: general register organization, stack organization, single-accumulator organization, instruction formats, addressing modes, data transfer and manipulation, program control, CISC and RISC computers, and examples of CISC and RISC processor. Computer arithmetic: addition, subtraction, multiplication and division algorithms, and floating point arithmetic operations. Input-output organization: input-output interface, asynchronous data transfer, priority interrupt, DMA, IOP, and serial communication. Memory organization: memory hierarchy, main memory, auxiliary memory, associative memory, cache memory, virtual memory. Student will be trained on the available software such as: MultiSim 6.01, Mentor graphics and Xilinx software product in addition with the products and components of Heathkit educational systems such as Microprocessor modules (EWS 3800 microprocessor modules) beside to the Xilinx product and components for FPGA technology.

Course Objectives:

The student are introduced to

1. Understand the architecture and organization of computing systems and electronic computers, study the program execution, instruction format and instruction cycle.
2. Design a simple computer using hardwired and micro programmed control methods, study the basic components of computer systems besides the computer arithmetic, and understand input-output organization, and memory organization and management.

Grading:

- **Mid-term**: 20%
- **Project**: 20%
- **Quizzes**: 10%
- **Final**: 40%
- **Lab**: 20%
- **Participation**: 10%

Text Book:


Reference Book:


Course Description:

In this course, the student will study the basic concepts of operating systems (OS), including:

- OS Overview (objectives, functions, evolution of OS, characteristics of modern OS).
- Process description and control (process definition, process states, process description and process control).
- Threads (definition, why use thread, relationship between processes and threads).
- Microkernel (benefits of microkernel organization, microkernel design).
- Uni-processor scheduling (types of scheduling, short term scheduling criteria, scheduling algorithms).
- Memory management (memory management requirements, loading programs into main memory -fixed partitioning, dynamic partitioning, simple paging, simple segmentation-).
- Virtual memory (paging, segmentation, combined paging and segmentation).
- Operating system software (fetch policy, placement policy, replacement policy, resident set management, cleaning policy, load control).
- I/O management and disk scheduling (I/O devices, organization of I/O function, I/O buffering, disk I/O).
- File management (file management system, file organization and access, file directories, secondary storage management).

Course Objectives:

1. Understand the basic concepts underlying operating systems and how a typical operating system works.
2. Describe the functions and design of operating systems.
3. Understand the main concept behind traditional (non-distributed) operating systems.
4. Analyze and explain the Algorithms used in Virtual Memory Management.
5. Discuss the algorithms used in I/O and File Management.

Grading:

- Mid-term: 20% 
- Project: 20% 
- Quizzes: 10% 
- Final: 40% 
- Lab.: 20% 
- Participation: 10%

Text Book:


Reference Books:

Course Name: Digital Logic
Course Code: CNET 111
Credit Hours: 3
Contact Hours: Lec. 2, Lab. 2, Tot. 4

Track
- Core [ ]
- Major [X]
- Elective [ ]

Level: 6th Level
Prerequisite

Course Description:
Digital Systems: digital computer and digital systems, binary, decimal, octal and hexadecimal number systems, number base conversion, complements, signed and unsigned numbers, binary codes, binary storages and registers, and binary logic. Boolean algebra and logic gates: basic definitions, axioms definitions of Boolean algebra, basic theorem and properties of Boolean algebra, Boolean functions, canonical and standard forms, logic operations, and digital logic gates. Simplification of Boolean functions: the map methods, product of sum simplification, NAND and NOR implementation, and the tabulation method. Combinational logic circuits: adders, sub tractors, decoders, encoders, multiplexers, de-multiplexers, look-up table, function implementation using multiplexers/decoders and memories. Sequential logic circuits: flip-flops, synchronous and asynchronous circuits, counters (types of counters), registers, memories, design of counters, design of sequential circuits, analysis of counters, and analysis of sequential circuits. Analog-to-digital converters and digital-to-analog converters. Programmable logic devices (PLD): PLA, PAL and FPGA. Student will be trained on the available software such as: Circuit maker 2000, EWB50a, and MultiSim 6.01 in addition with the products and components of Heath kit educational; systems (EWS-3700 analog modules).

Course Objectives:
1. The student are introduced to
2. Study the fundamentals of digital systems, Boolean algebra, and logic expressions, simplify and implement Boolean functions using elementary logic gates,
3. Study the combinational and sequential digital circuits,
4. Design and analysis of combinational and sequential logic circuits, and study briefly the advanced mask and field programmable logic devices.

Grading
- Mid-term [X] 20%
- Project [ ] 20%
- Quizzes [X] 10%
- Final [X] 40%
- Lab. [ ] 20%
- Participation [X] 10%

Text Book:

Reference Books:
## Course Description

**Course Code:** INFS 241  
**Credit Hours:** 3  
**Contact Hours:** 2 Lec, 2 Lab, 4 Tot

### Course Name: Internet Technology

<table>
<thead>
<tr>
<th>Track</th>
<th>Core</th>
<th>Major</th>
<th>Elective</th>
</tr>
</thead>
<tbody>
<tr>
<td>☒</td>
<td>☒</td>
<td>☒</td>
<td>☒</td>
</tr>
</tbody>
</table>

### Level: 6th Level

#### Prerequisite

Introduction to current technology on the Internet. Students will learn about existing Internet applications, such as email, instant messaging, file transfer, secure communications, and the Web, and will be introduced to related tools and protocols. Users will complete the course with a basic understanding of the Web-publishing process and methods used to locate authoritative information on the Internet. Webpage design, Internet security and emerging/declining technologies on the Internet will also be discussed. Prerequisite: basic computer literacy recommended.

### Course Objectives:

After completing this course, students will be able to:

1. Understand the Web-publishing process;
2. Use basic XHTML code to develop a simple homepage;
3. Use a UNIX operating system for elementary procedures related to website management.
4. Use client software for email, FTP, SSH, VPN and the World Wide Web and be able to find and retrieve authoritative information on the Internet using various resources and search engines.
5. Use a Web forum.
6. Use various methods of online chat.
7. Understand the underlying communication protocols and physical infrastructure of the Internet.
8. Understand basic security issues and solutions related to the Internet.
9. Discuss the technical requirements for transmission of text, voice and video data.
10. Download, decompress, and create secure archives (ZIP files).
11. Discuss technology trends and consider emerging (and declining) technologies related to the Internet.

### Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Final</th>
</tr>
</thead>
<tbody>
<tr>
<td>☒</td>
<td>☑</td>
<td>☐</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☐</td>
<td>☐</td>
<td>☑</td>
</tr>
</tbody>
</table>

- 20% Mid-term
- 20% Quizzes
- 40% Final
- 20% Lab
- 20% Participation

### Text Book:

Perry, Schneider: New Perspectives on The Internet Course Technology, 6th Edition  
ISBN 1-4188-6071-9

### Reference Books:

Course Description:
This course offers to students who desire to get deeper knowledge and specialize in AI. This course is about the theory and practice of Artificial Intelligence. It covers modern techniques for computers to represent task-relevant information and make intelligent decisions (i.e. satisfying optimal) towards the achievement of goals. It covers also some advanced topics in artificial intelligence and cognitive modelling. Recent offerings of the course have focused on Computational Learning Theory, Machine Learning, Data Mining, and Knowledge Discovery, Neural and Evolutionary Computation, and Multi-Agent Systems basics. Near the end of the course we will spend several lectures learning about and discussing some important current application areas of AI. The target audience are students who are interested in Artificial Intelligence or its applications to problems in other areas of computer science (e.g., computer networks, intelligent databases, information retrieval, program synthesis, automated discovery, automated design, robotics, operating systems, parallel and distributed computing, application of Intelligent Systems in Engineering. Students will also be trained on some popular AI software like: CLIPS, Experts, BrainCel, Exsys, Evolver, GenJam, and some relevant AI toolboxes of MATLAB.

Course Objectives:
We expect that by the end of the course students will:
1. have a thorough understanding of the algorithmic foundations of AI, how probability and AI are closely interrelated,
2. get modern techniques for computers to represent task-relevant information and make intelligent decisions towards the achievement of goals,
3. get acquainted with new trends in AI: Computational Learning Theory, Machine Learning, Knowledge Discovery, Evolutionary Computation, …. 
4. cover applications to problems in other areas of computer science: computer networks, intelligent databases, parallel and distributed computing, application of Intelligent Systems in engineering, etc...
5. have a strong appreciation of the big-picture aspects of developing fully autonomous intelligent agents.

Grading:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mid-term</td>
<td>20%</td>
</tr>
<tr>
<td>Project</td>
<td></td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
</tr>
<tr>
<td>Lab</td>
<td>20%</td>
</tr>
<tr>
<td>Participation</td>
<td>10%</td>
</tr>
</tbody>
</table>

Text Book:

Reference Book:
Course Description

**Department of Computer Science**

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Algorithms &amp; Data Structures 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course Code</strong></td>
<td>COMP 222</td>
</tr>
<tr>
<td>Credit Hours</td>
<td>3</td>
</tr>
<tr>
<td>Contact Hours</td>
<td>Lec 2 Lab 2 Tot 4</td>
</tr>
<tr>
<td>Track</td>
<td>Core</td>
</tr>
<tr>
<td></td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>Elective</td>
</tr>
<tr>
<td>Level</td>
<td>6th Level</td>
</tr>
<tr>
<td>Prerequisite</td>
<td>COMP 221</td>
</tr>
</tbody>
</table>

**Course Description:**
The course teaches basic techniques for data abstractions, access algorithms, and manipulation of the abstract structure, as well as an introduction to complexity analysis of space and time allocation in implementing the algorithms. The topics covered are: Abstract Data Type Concept, Linear Data Model: arrays ad dynamic lists, hierarchical Data model, Binary tree, Heap, Binary Search Tree, AVL-Tree, Graph Model, Hashing, Searching and Tracing Algorithm, Garbage Collection and Memory Management.

**Course Objectives:**
1. This course introduces concepts and techniques of data structure and Analysis of Algorithm.
2. Implement various data structures and their algorithms, and apply them in implementing simple applications.
3. To analyze simple algorithms and determine their efficiency using big-O notation.
4. To apply the knowledge of data structures to other application domains like data compression and memory management.
5. Use big O, omega, and theta notation to give asymptotic upper, lower, and tight bounds on time and space complexity of algorithms.

**Grading:**
- **Mid-term:** 20%
- **Project:**
- **Quizzes:** 10%
- **Final:** 40%
- **Lab:** 20%
- **Participation:** 10%

**Text Book:**

**Reference Books:**
1. Adam Drozdek “Data structure & Algorithm in C++” Thompson Asia Pvt. Ltd (Singapore).
## Course Description


## Course Objectives:

Making the student understand the main concepts of:
- Fundamentals of data communication, and transmission media
- Fundamentals of Networking, Network protocols, and networking devices
- Circuit and packet switching
- Data link layer concepts and control
- Error detection and correction techniques
- Networking and internetworking devices
- Net centric computing circuits.

## Course Code: CNET 331

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Computer Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
</tr>
<tr>
<td>Contact Hours</td>
<td>Lec. 2 Lab. 2 Tot. 4</td>
</tr>
<tr>
<td>Track</td>
<td>Core CS Major CNET</td>
</tr>
<tr>
<td>Level</td>
<td>7th Level</td>
</tr>
<tr>
<td>Prerequisite</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Grading

<table>
<thead>
<tr>
<th></th>
<th>20%</th>
<th>10%</th>
<th>20%</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mid-term</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
<td>Lab.</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>Project</td>
<td>20%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quizzes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Text Book:

## Reference Books:

Course Description:
An introduction to programmer model of computer organisation using assembly and machine language. Process of translation from high level language to machine instructions. Number representation, computer arithmetic, instruction sets, I/O interfacing, I/O interrupts, and programming interrupts. Laboratory exercises involve detailed study and interfacing of an ARM based microprocessor hardware and software system.

Course Objectives:
At the end of the course the student should:

1. know basics of microprocessor-based Systems
2. know basics of assembly language
3. know the process of compilation from high level language to assembly language to machine language.
4. know interaction between hardware and software, i.e. ‘interfacing’.

Text Book:

Reference Books:
Course Description

Course Name: Analysis & Design of Algorithms
Course Code: COMP 323

Credit Hours: 3
Contact Hours: Lec. 2, Lab. 2, Tot. 4

Track: Core CS, Major IS, Elective CNET

Level: 7th Semester
Prerequisite: COMP 222

Course Description:
This course includes both algorithms and complexity, the purpose of the first part is to provide students with techniques for designing and analyzing algorithms. The algorithm design paradigms, such as Brute-force; greedy; divide-and-conquer; backtracking; branch-and-bound; heuristics; pattern matching and string/text algorithms; numerical approximation. Students should acquire some understanding on design techniques and algorithms that address an important set of well-defined problems: Depth- and breadth-first traversals; shortest-path algorithms (Dijkstra's and Floyd's algorithms); transitive closure (Floyd's algorithm); minimum spanning tree (Prim's and Kruskal's algorithms); topological sort. In addition, the course will provide different complexity characteristics (P and NP classes, NP-completeness, reduction techniques). Finally, advanced algorithm analysis is studied (randomized algorithm, dynamic programming and combinatorial optimization).

Course Objectives:
After completing this course the student will:
1. Understand the analysis and design of algorithm.
2. Understand and describe divide-and-conquer as a strategy for designing algorithms.
3. Understand and describe the dynamic programming strategy for algorithm design.
4. Understand and describe the greedy strategy for algorithm design.
5. Give a basic intuitive definition of the computational complexity classes P, NP, NP-Hard, and NP-Complete.

Grading

| Grading  | Mid-term | 20% | Project |  | Quizzes | 10% |
|----------|----------|-----|---------| |         |     |
| Final    |          | 40% | Lab.    | 20% | Participation | 10% |

Text Book:

Reference Books:
1. Cormen, Leiserson, Rivest, Introduction to Algorithms: Prentice Hall of India
   By David Harel
   Addison-Wesley, 1992
   ISBN 0-201-50401-
Course Description

Department of Computer Science

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

Course Code: COMP 324

Course Name: Computer Data Security & Privacy

Credit Hours: 3

Contact Hours: Lec 2, Lab 2, Tot 4

Track
- Core
- Major
- CS
- IS
- Elective
- CNET

Level: 7th Level

Prerequisite

Course Description:
This course presents relevant aspects of computer security and privacy. It includes the following topics: Security fundamentals: concepts and principles, vulnerability, threat models, attacks to computer systems. Data base and networks, cryptography: notion of public key, private key. Cryptology, authentication, digital signatures, key management and cryptography protocols, building secure systems, security in operating systems: protection mechanisms, OS services, access control, UNIX and windows NT security, network security: architecture and standards, authentication, access control, confidentiality, integrity, network management, internet security, firewalls, DNS and routers, computer security policy and procedures, and ISO security standards. Students will also be trained to use some specific security software like: Spector, Privilege, Esafe, Etoken, Hardlock.

Course Objectives:
The main objectives of this course are:
1. To introduce students to concepts and principles of security, cryptographic systems, and protection mechanisms of operating systems,
2. To make the student able to design and build secure system and secure networks relying on well-known security software,
3. To make the student able also to manage networks,
4. To introduce the student to standard computer security policies and procedures.

Grading
- Mid-term: 20%
- Project: 10%
- Quizzes: 10%
- Lab: 20%
- Participation: 20%
- Final: 40%

Text Book:

Reference Books:
## Course Description

**Course Name:** Data Warehousing & Data Mining  
**Course Code:** INFS 323

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>Contact Hours</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Lec 2</td>
<td>Lab 2</td>
<td>Tot 4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Track**  
- □ Core  
- [ ] Major  
- [ ] COMP  
- [ ] INFS  
- [ ] CNET  
- □ Elective

**Level**  
8th Level

**Prerequisite**  
INFS 221

### Course Description:
Data mining concepts: why data mining?, cycles of data mining, the various cycles in practice, data mining methodology, measurement of the effectiveness of data mining. It will introduce various data mining techniques: the market based analysis, clustering, link analysis, decision trees, artificial neural networks, genetic algorithms, data mining and the corporate data warehouses, OLAPs, and choosing the right tool for the job, putting data mining to work. The course introduces also data warehouse concepts: Gradual changes in computing, dynamic reports, data marts, operational Data stores, and data warehouse cost-benefit analysis. Some other concepts are described such as: Warehousing strategy, warehouse management and support processes, data warehouse planning, data warehouse implementation, data warehouse maintenance and evolution, warehouse applications and warehouse software, and recent warehouse trends. Students will also be trained on some well-known data mining software like: Matryx98, Cart, Megaputer PolyAnalyst, Knowledge Access, Cognos Power Play.

### Course Objectives:
After completing this course, the student should be able to:
1. Understand Data Warehouse and OLAP technology for data mining: Data preparation, data mining primitives, languages, and system architectures,
2. Make mining association with rules in large databases, do classification and prediction (with various techniques: Cluster analysis, neural nets, genetic algorithms...),
3. Develop and understand data mining applications and trends of data mining,
4. Deal with Warehousing strategy, warehouse management and support processes,
5. Have skills in data warehouse planning, data warehouse implementation, data warehouse maintenance and evolution,
6. use some warehouse software related to some warehouse applications, and be acquainted with recent warehouse trends.

### Grading:
- [ ] Mid-term 20%
- [ ] Project 10%
- [ ] Quizzes 15%
- [ ] Final 40%
- [ ] Lab 10%
- [ ] Participation 5%

### Text Book:

### Reference Books:
**Course Description:**

This course presents a review of the basic concepts of the operating systems (processes and threads, process states, process scheduling), an introduction to distributed systems (what is a distributed system? What is a real-time system? What is a parallel system?) And some sample distributed application. This course includes both concurrency and distributed systems; the purpose of the first part is to provide students with some understanding on mutual exclusion and synchronization (principles of concurrency, mutual exclusion - software and hardware approaches-, semaphores, monitors, message passing, readers/writers problems) and on deadlock and starvation (principles of deadlock, deadlock prevention, deadlock detection, deadlock avoidance, dining philosophers problem). The distribution concerns: memory management (review of centralized memory management, simple and shared memory model, distributed shared memory and memory migration), distributed process management (distributed scheduling algorithm choices, scheduling algorithm approaches, coordinator elections and orphan processes) and distributed file systems (distributed name service, distributed file service, distributed directory service and NFS. X.500). Students will be trained on some software tools such as: Unix, WINDOWS.NT, and CORBA.

**Course Objectives:**

1. Understand the concepts of the distributed systems.
2. Describe the principle of the concurrency and the approaches to achieve mutual exclusions.
3. Investigate the principle of the deadlock and the approaches used in handling it.
4. Analyze and explain the algorithms used in shared memory management.
5. Be able to understand the algorithms used in distributed process management.
6. Discuss the algorithms used in distributed file systems.

**Grading:**

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Final</th>
<th>Lab</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%</td>
<td></td>
<td>10%</td>
<td>40%</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

**Text Book:**


**Reference Books:**

**Course Description:**

This course presents both technical and managerial software engineering problems, before studying in depth some life cycle phases, student should understand fundamentals aspects such as: the software life cycle and its phases, the software development models and different specialized systems. In this course, emphasis will be placed on the latest software engineering life cycle phases: the software validation (testing, testing levels, test case generation, black-box and white-box testing techniques) and the software evolution (maintenance, reuse, reengineering, legacy systems). The main topics covered in software validation include formal methods (formal specification languages, formal verification). The construction process considered as one of the inputs to testing process is not studied here; students acquire understanding of coding at programming courses. Students should acquire some fundamentals of software project management: team management; project scheduling; risk analysis; software configuration management; project management tools. Students will be trained on some software tools such as: Rational Unified Process, and Z Language.

**Course Objectives:**

1. Understand software validation: Validation planning; testing fundamentals, including test plan creation and test case generation; formal methods concepts, formal verification.
2. Understand software evolution: Software maintenance; characteristics of maintainable software; reengineering; legacy systems; software reuse.
3. Work in team, take a project, test and maintain a small or medium-scale system.
4. Prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, and project risk.
5. Make use of available case tools.

**Grading:**

<table>
<thead>
<tr>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>20%</td>
<td></td>
<td>10%</td>
<td></td>
</tr>
<tr>
<td>Final</td>
<td>Lab</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

**Text Book:**


**Reference Books:**

Course Description:

Fundamentals of computer graphics programming. Graphics hardware and software standards. 2D geometric primitives and raster images. 3D object representations. Data structures, algorithms, and the graphics pipeline. Graphical user interfaces. Underlying concepts in computer graphics systems, including games, animation, modelling, rendering, and paint systems.

Course Objectives:

After completing this course the student will:
1. To encapsulate event-driven graphical objects.
2. Design and implement low level object-oriented code:
   - encapsulate event-driven graphical objects,
   - perform homogeneous transformations,
   - parameterize curves for controlling object trajectories and improving rendering,
   - designing new viewing models,
   - cast simple shadows,
   - perform culling and clipping of polygons,
   - implement movable light sources with object shadows,
   - perform shading, texture map and Animate scenes with multiple interacting objects.

Grading

<table>
<thead>
<tr>
<th></th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%</td>
<td></td>
<td>10%</td>
</tr>
<tr>
<td>Final</td>
<td>40%</td>
<td>Lab.</td>
<td>Participation</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td></td>
<td>10%</td>
</tr>
</tbody>
</table>

Text Book:

Reference Books:
Course Description

Department of Computer Science
Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

Course Code: INFS 336

Course Name: Human Computer Interaction

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Lec 2, Lab 2, Tot 4</td>
</tr>
</tbody>
</table>

Track: Core ☑ Major ☑ Elective ☑

Level: 8th Level

Prerequisite: 8th Level

Course Description:
This course provides an overview and introduction to the field of HCI. It introduces students to tools, techniques, and sources of information about HCI and provides systematic approach to design. The course increases awareness of good and bad design through observation of existing technologies, and teaches the basic skills of task analysis, and analytic and empirical evaluation methods. The student will be acquainted with the whole design process: HCI in the design process, design rules, implementation support, evaluation techniques, universal support, etc...He also studies some relevant models and theories: cognitive models, communication models, task analysis, dialog notations and design, modelling rich interaction, etc ...Final chapters will cover some alternative realities, multimedia, global information systems, and the Web. Students will also participate in a laboratory where they will practice HCI techniques in an independent, self defined project. Students will be trained on some HCI software like: AlphaUIMS, SuperCard, ISA dialog Manager, InterMaphics.

Course Objectives:
After completing this course the student will:
1. Acquire some useful HCI techniques in practice,
2. Cover the latest topics in multimedia, global information systems, and the web-based models for rich interaction,
3. Increase coverage of social and contextual models and theories related to HCI design processes.
4. Be acquainted to new topics like: interaction design, universal access, and rich interaction.

Grading:
- ☑ Mid-term 40%
- ☑ Project 40%
- ☑ Quizzes 10%
- ☑ Lab 40%
- ☑ Participation 10%

Text Book:

Reference Books:
Course Description:
This course presents aspects of mobile computing. It shows how mobile devices like PDA's, notebooks, and mobile phones can work with fixed network computers in building files, database and web client-server systems for achieving the goal of computing in wireless mobile environment anytime and anywhere. The technologies involved to realize such a system are covered as well as the fundamental concepts of mobile computing. This course focuses on data management in mobile computing environment, and in particular in distributed mobile file, database, web client-server, and object systems. Students in this course are assigned a project to demonstrate their ability to handle mobile computing operations. The student will use some systems like Aglet and Concord systems.

Course Objectives:
Making the student to:
study software architecture in a mobile computing environment, understand system support for dealing with disconnected operations, weak connectivity, broadcast and mobility, understand information representation, study dissemination and management, Location management, caching and replication, and study concurrency and recovery and many other important related issues.

Grading
- ☒ Mid-term 20%
- ☐ Project
- ☒ Quizzes 10%
- ☒ Final 40%
- ☒ Lab 20%
- ☒ Participation 10%

Text Book:

Reference Books:
<table>
<thead>
<tr>
<th>Course Name</th>
<th>Software Project Management</th>
<th>Course Code</th>
<th>INFS</th>
<th>433</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
<td>Lec 2</td>
<td>Lab 2</td>
<td>Tot 4</td>
</tr>
<tr>
<td>Track</td>
<td>[ ] Core [ ] Major [ ] Elective</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>9th Level</td>
<td>Prerequisite</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course Description:**
This course will start by reviewing management and project management principles. It continues by studying the system life cycle: analysis (requirements determination), designing, implementation; system and database integration issues; network management; project tracking techniques, metrics, and system performance evaluation; managing expectations of managers, clients, team members, and others; determining skill requirements and staffing; cost-effectiveness analysis; reporting and presentation techniques; management of behavioural and technical aspects of the project; change management. This course teaches about software tools for project tracking and monitoring. Team collaboration techniques and tools. In addition, this course will introduce students to project-time scheduling methods. Students will be trained on some software tools such as: MS project, CLARITY, Visual Studio 2005, JIRA, and VA Software.

**Course Objectives:**
1. To teach students skills needed to design a project development and implementation plan
2. To develop skills in use of project management tools and methods within the context of an information system project
3. To initiate, design, implement, and discuss project close down
4. To determine requirements and specifications for multi-user information system based on a database
5. To present and explain the evolving leadership role of information management in organization
6. To examine the process for development of information system policies, procedures, and standards in the organization.
7. To discuss outsourcing and alternate implementations of IS Function

**Grading**
- Mid-term 20%
- Project 10%
- Quizzes 15%
- Final 40%
- Lab 10%
- Participation 5%

**Text Book:**

**Reference Book:**
Course Description: This course covers the design and implementation of the technologies used to implement interactive multimedia applications such as streaming video playback, video conferencing, interactive television, video editing, and hypermedia authoring. Fundamentals of human perception, digital media representations, compression and synchronization are covered. Implementation technologies including hardware architectures for media processing (e.g., processor, bus, and input/output devices), OS support, multimedia systems services, network architectures and protocols, and distributed programming services are also discussed.

- Introduction to Multimedia, Graphics for Multimedia and the World Wide Web
- Website Design, Advanced Website Design and Development, Streaming Media for the Web
- E-Commerce on the Web, Introduction to DHTML, Introduction to XML, Rich Web Development with Flash" MX (Advanced)

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

- Produce and integrate multimedia rich elements, such as images, sound, video and animations
- Create complex interactive applications, through programming and/or scripting, particularly for the Web
- Help people by allowing them to avail of the best technological solutions.

Text Book:

Reference Book:
Niderest, J (2001): Web Design in a Nutshell
Course Description:
This course presents an introduction to compilers (phases –analysis, synthesis, table of symbols-, architecture and tools), lexical analysis (lexical tokens, regular expressions, regular expressions, finite automata -DFA and NFA-, lexical analyzer generator), parsing (context-free grammar, top-down parsing -recursive descent and LL(K)-, bottom-up parsing –operator precedence, SLR, LALR-, syntax analyzer generator -YACC/Bison-), semantic analysis (semantic actions, symbol table, type checking), intermediate code generator (IC language, run-time environment -storage organization-, registers using), code generation (target machine language), code optimization (optimization techniques, loop optimization), dealing with object-oriented languages and functional languages (visibility, typing, class belongingness).

Course Objectives:
1. Provide the students with a clear overview of how to construct grammar for given programming language.
2. Be able to understand the concepts of Lexical analysis syntax analysis, and semantic.
3. Understand the concepts of parsing and code optimization.
4. Understand the main scope of object oriented languages.
5. Design and build a small-scale compiler.

Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Final</th>
<th>Lab</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25%</td>
<td></td>
<td>20%</td>
<td>40%</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

Text Book:

Reference Book:
Course Description

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

Course Code: COMP 495
Course Name: Final Project
Credit Hours: 3
Contact Hours: Lec 3, Lab 0, Tot 3

Track: ☑ Core
☐ Major
☐ Elective

Level: 10th Level

Prerequisite

Course Description:
Project implementation course offers students an opportunity to assemble their knowledge acquired throughout their BS curriculum to realize a final project. This would require them to gather information about the proposed subject and realize a final report as well as to develop a system practically. At this stage, students must carry on all phases of system analysis, design, and implementation of the subject already defined in the preceding course (Project proposal), and under the supervision of the same supervisor. At the end of the semester, students are asked to make an oral presentation with the presence of faculty members as referees.

Course Objectives:
The objectives of this course are to help students learn how to:
1. Acquire new knowledge and skills and apply them in a real life project.
2. Implement the selected solution
3. Learn how to test the implementation
4. Readjust and make the necessary changes on the implemented system.
5. Make and write the necessary documentation
6. Present project work.

Grading:
☒ Weekly Report  40%  ☑ Final Project  60%

Text Book:
There is no text book for this course

Reference Books:
# Course Description

Department of Computer Science  
Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Artificial Neural Networks</th>
<th>Course Code</th>
<th>COMP</th>
<th>442</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
<td>Contact Hours</td>
<td>Lec.</td>
<td>Lab.</td>
</tr>
<tr>
<td>Track</td>
<td>□ Core</td>
<td>✓ Major</td>
<td>□ IS</td>
<td>✓ CNET</td>
</tr>
<tr>
<td>Level</td>
<td>10th Level</td>
<td>Pre requisite</td>
<td>COMP 241</td>
<td></td>
</tr>
</tbody>
</table>

## Course Description:
AI Definitions, Knowledge representation, Search techniques, Connectionist neural networks, learning and adaptation, self-organization, fuzzy set theory and fuzzy logic, intelligent agents, genetic algorithms, Internet applications.

## Course Objectives:
- To understand AI Definitions.
- To understand Connectionist neural networks.
- To understand fuzzy set theory and fuzzy logic.

## Grading:
<table>
<thead>
<tr>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>20%</td>
<td></td>
<td>10%</td>
<td></td>
</tr>
<tr>
<td>Final</td>
<td>Lab.</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>40%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Text Book:

## Reference Books:
Course Description: Computer Ethics & Society

Course Code: INFS 452

Credit Hours: 3

Contact Hours: 2 Lec, 2 Lab, 4 Tot

Track: Core, Major, Elective

Level: 10th Level

Prerequisite: COMP, INFS, CNET

Course Description:
Introduction, Viruses, Worms, and Trojan Horses, Phreaks and Hackers, Denial-of-Service Attacks, Online Voting.
Introduction, Data-Entry or Data-Retrieval Errors, Software and Billing Errors, Notable Software System Failures, Computer Simulations, Software Engineering, Software Warranties.

Course Objectives:
Upon successful completion of this course, students will be able to:
1. Students should acquire a broad perspective on the social and ethical impacts and implications of information technology.
2. Students should acquire specific knowledge about major issues in several different areas of the field of Computer Ethics.
3. Students should acquire in-depth knowledge of at least one significant ethical issue generated by information technology.
4. Students should develop skills in clarifying and ethically analyzing realistic cases that involve information technology.
5. Students should exercise and improve their skills in critical and analytical writing.

Grading:
- Mid-term: 20%
- Project: 10%
- Quizzes: 15%
- Final: 40%
- Lab: 10%
- Participation: 5%

Text Book:

Reference Book:
### Course Description:
This course should be taken by those students completed 90 credit hours. The course period is 12 weeks long and must be during summer of the third academic year in. Students must be oriented in one of the companies, and well supervised so to accomplish correctly this training. The course must constitute a link between the theoretical and scientific academic background and the work environment. It provides a better understanding and a clear view of the real-world work environment. It provides also students complementary knowledge and training such as facing and dealing with real-world problems, being trained to work in team-works. After completing the summer training, students must submit a report. An oral exam is held by a committee consisting of both faculty members and outsider supervisors.

### Course Objectives:
1. Develop student skills using practical applications.
2. Acquaintance the work environment.
3. Prepare the students to transfer from learning environment to work environment.
4. Acquaintance the applied work systems.
5. Understand mechanism of different applications.
6. Understand the attitude and the manner of the work.
7. Compare the studying courses with real world.

### Grading
Pass or Fail

### Text Book:

### Reference Books:
# ELECTIVES COURSES IN COMPUTER SCIENCE

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Advance Algorithm Design and analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Code</td>
<td>COMP 425</td>
</tr>
<tr>
<td>Credit Hours</td>
<td>3</td>
</tr>
<tr>
<td>Contact Hours</td>
<td>Lec. 2  Lab. 2  Tot. 4</td>
</tr>
<tr>
<td>Track</td>
<td>Core  ☑  Major  ☐  Elective  ☐</td>
</tr>
<tr>
<td>Level</td>
<td>9th Level</td>
</tr>
</tbody>
</table>

**Course Description:**
Algorithm design using dynamic programming, randomization and greedy methods. Analysis using recurrence relations and amortization. String and network algorithms. NP-completeness, reductions and approximation algorithms. Review of divide-and-conquer design and asymptotic notation. Prerequisite: At least a C- grade in CSC 130 and full CSC major status.

**Course Objectives:**
This course introduces advanced topics in algorithm design and analysis, including:
1. Analysis of recursive functions and functions requiring amortized analysis.
2. Use of algorithms to solve problems for which they were not designed.
3. Algorithm design and analysis techniques: randomized, dynamic programming, greedy, divide-and-conquer.
4. NP-Completeness including reductions.

**Grading:**
- Mid-term: 20%
- Project: ☐
- Quizzes: 10%
- Final: 40%
- Lab.: ☑ 20%
- Participation: 10%

**Text Book:**

**Reference Books:**
1. Cormen, Leiserson, Rivest, Introduction to Algorithms Prentice Hall of India

ISBN 0-201-50401-
**Course Description:**
The course begins by introducing an introduction to Web Technology, TCP/IP, Protocols, Telnet, Electronic mail (E-Mail), File Transfer Protocol (FTP), World Wide Web (WWW), Domain Name System (DNS). Next the course covers Hypertext Mark-up Language HTML, Tags, Anchors, Backgrounds, Images, Web Page Structure, Hyper linking, Lists, Character Formatting.

Dynamic Web Pages HTML/DHTML, Forms, Client-side Forms, Java script expressions, Control Flow and Functions, Java Script Objects, Java Script Forms, Cookies, History, Location, XML, CGI Scripting with Perl. are introduced. The course covers also Active Server Pages & Servlets ASP objects, Applications, Response, Server, Session, Forms, Query, Strings, Cookies and connectivity with database.

**Course Objectives:**
The objective of the course is to provide an understanding of technology used for building WEB. The course gives knowledge right from building of WEB to making business on WEB. It also gives a comprehensive coverage of HTML, Java Script and CGI/Pert.

**Grading:**
- Mid-term: 20%
- Project: 0%
- Quizzes: 10%
- Final: 40%
- Lab.: 20%
- Participation: 10%

**Text Books**
1. Kriss Jamsa & Konnad King "HTML & Web Design" TMH
2. Achyut Godbole "Web Technology" TMH

**References**
1. Douglas E. Comer "Internetworking With TCP/IP" Vol. 1, Pearson education
Course Name: Java Programming  
Course Code: COMP 417

<table>
<thead>
<tr>
<th>Credit Hours</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Lec 2, Lab 2, Tot 4</td>
</tr>
</tbody>
</table>

Track: CS  
Level: 9th Level  
Prerequisite:  

Course Description:
- Introduction to Java Applications
- The Java Programming Environment
- Fundamental Programming Structures in Java
- Objects and Classes in Java
- Inheritance
- Interfaces and Inner Classes
- Exceptions and Exception Handling in Java
- Streams and Files
- Applets
- Swing, Graphical User Interface Components
- Multithreading
- Java Database Connectivity
- Servlets

Course Objectives:
- Demonstrate Object Oriented Programming concepts, including composition, inheritance, and polymorphism, using the Java programming language;
- Install and run the Java runtime environment;
- Develop, compile, and run Java applications;
- Design, build, and run Java GUI applications using Swing and AWT;
- Develop simple web applications using the J2EE framework;

Grading:
- Mid-term 20%
- Final 40%
- Project 20%
- Lab 20%
- Quizzes 10%
- Participation 10%

Text Book:
Java How to Program, Seventh Edition by Harvey M. Deitel and Paul J. Deitel

Reference Books:
1) D. Flanagan, "Java in a Nutshell", O'Reilly.
3) Java 2: The Complete Reference, Fifth Edition by Herbert Shield
### Course Description:
The application of operating system principles to the design and implementation of a Multitasking operating system. Students will write an operating system for a computer platform. Topics include: scheduling of processes, control and allocation of computer resources, and user interfacing. Prerequisite: At least a C- in CSC 139 and full CSC or CPE major status. Cross-listed as CPE 159; only one may be counted for credit.

### Course Objectives:
By coding a working kernel and other operating system components, to give students hands-on practical experience in the application of operating system principles covered in CSC 139, so they will have a solid and thorough understanding of the design and implementation issues surrounding the development of contemporary operating systems.

### Text Book:

### Reference Books:


Pentium Processor Register and Data Structure Summary, Intel Corporation.

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Operating System Pragmatics</th>
<th>Course Code</th>
<th>COMP 334</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
<td>Contact Hours</td>
<td>Lec. 2</td>
</tr>
<tr>
<td>Track</td>
<td>Core</td>
<td>Major</td>
<td>Elective</td>
</tr>
<tr>
<td>Level</td>
<td>7th Level</td>
<td>Prerequisite</td>
<td></td>
</tr>
</tbody>
</table>

Grading:

- **Mid-term**: 20%
- **Project**: 20%
- **Quizzes**: 10%
- **Final**: 40%
- **Lab.**: 20%
- **Participation**: 10%
Course Description:
The course begins by introducing Unix System kernel and utilities, File & Directories, Simple & Compound statements, Command libraries. Unix System Administration is also covered which includes File system mounting, System booting and shutting down, handling user account, terminals, printers and modems. The course also introduces different tools and debugger which include system Development Tool, Lex and M4, text formatting, tools troff, Debugger, Dbx, Adb and Ctrace. The course then covers Unix Shell Programming, which includes Bourne shell, Korn shell and C shell, Shell meta characteristics, Shell variable and scripts, integer and strings. The course ended by introducing Portability with C.

Course Objectives:
The aim of this course is to make the student familiar with the UNIX operating system and the shell programming in UNIX. This course also gives an idea about how one can communicate with others using UNIX and change the set up of UNIX operating system.

References
1- Kochen & Wood " Unix Shell Programming"
2- Fileder ,Hunter " Unix System Administration "

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Unix &amp; Shell Unix and Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Code</td>
<td>COMP 316</td>
</tr>
<tr>
<td>Credit Hours</td>
<td>3</td>
</tr>
<tr>
<td>Contact Hours</td>
<td>Lec. 2 Lab. 2 Tot. 4</td>
</tr>
<tr>
<td>Track</td>
<td>Core</td>
</tr>
<tr>
<td>CS</td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>7th Level</td>
</tr>
<tr>
<td>Prerequisite</td>
<td></td>
</tr>
</tbody>
</table>

Grading:
- Mid-term: 20%
- Project: 20%
- Quizzes: 10%
- Lab.: 20%
- Participation: 10%
Course Description:
This course is an introduction to Graph Theory and its applications, covering topics in Graph.
Isomorphism’s, Trees and its applications, Bipartite Graphs and Matching, Euler and Hamiltonian Graphs, Graph Coloring, Planar Graphs, Metrical Representations, Digraphs and networks, with numerous graph algorithms throughout.

Course Objectives:
The aim of this course is to development of algorithms to handle graphs.

Grading

<table>
<thead>
<tr>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>20%</td>
<td></td>
<td>10%</td>
<td>20%</td>
</tr>
</tbody>
</table>

Text Book:
2- Discrete Mathematical Structures, Mallik and Sen, Thomson

References:
1-Buckley and Lewinter, A Friendly Introduction to Graph Theory, 2003, Prentice Hall
2-Reinhard Diestel, Graph Theory, 3rd Edition 2005, Springer Verlag
Course Name: Real Time Systems
Course Code: COMP 435

Credit Hours: 3
Contact Hours: Lec. 2, Lab. 2, Tot. 4

Track
- Core
- Major
- Elective

Level: 9th Level

Course Description:
The course begins by introduction to real time systems, priorities, embedded systems, task classifications, requirements, deadlines. The course also introduces real time operating systems, task management, inter process communication. Characterizing real time systems and tasks is also covered by the course. Finally the course introduces task assignment, scheduling theory, fixed and dynamic priority, fault tolerance real time system and issues in real time software design.

Course Objectives:
The objective of this course is to make the student familiar with the concepts of real time systems, their characteristics, requirements and relationship with embedded systems. The course also includes programming languages and tools used to design the real time systems.

Grading
- Mid-term 20%
- Project
- Quizzes 10%
- Final 40%
- Lab. 20%

References
1- Krishna C.M "Real Time Systems" McGraw Hill
2- Jane W. S. Lin "Real Time Systems" Person education Asia
### Course Description:
The course begins by introduction to
The .Net framework, common language routine, common type system, the base class library, the .Net class library and just in time compilation. The course also covers the data types, identifiers, variables, constrains, object oriented concepts, arrays and strings. The next chapter concerned with C# using libraries, namespace system, I/O, multi threading networking and sockets, data handling. Advanced feature using C# is introduced in the next chapter, which includes web services, Windows services, distributed application in C# and interface with C#.

### Course Objectives:
The objective of the course is to develop the awareness in the student about using visual programming using C#. By the end of the course the student will be trained in the creation of visual programming particularly in C#. Net platform

### Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight</td>
<td>20%</td>
<td>40%</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

### References

1- Shibi Pamikkar & Kumar Sanjeev " C# with . Net Frame Work 
2- Shildt, " C# The Complete Reference", TMH
Course Description

Department of Computer Science

Faculty of Computer Science and Information Systems, Jazan University, Jazan, KSA

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Computer Vision</th>
<th>Course Code</th>
<th>COMP</th>
<th>462</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Hours</td>
<td>3</td>
<td>Contact Hours</td>
<td>Lec</td>
<td>Lab</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Track</td>
<td>[] Core</td>
<td>Major</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[CS] Major</td>
<td>Elective</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>[9th Level]</td>
<td>Prerequisite</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prerequisite</td>
<td>[9th Level]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Course Description:

Algorithm development for automatic interpretation of the three-dimensional world that is captured in a set of images; cameras and image formation; color; keypoint and edge detection; perceptual grouping; segmentation; shape representation; texture; object recognition; optical flow; motion estimation and tracking; and 3D scene reconstruction from motion and stereo etc.

Course Objectives:

Making the student to:
Capture digital images, and master low-level, mid-level and high-level computer vision techniques, such as noise cleaning, feature extraction, template matching, and depth recovery from stereo and object recognition.
To develop a practical appreciation of the main algorithms and methods for image processing, image segmentation, 3D scene analysis and object recognition.
Become proficient with computer skills for the analysis of digital images.

Grading

<table>
<thead>
<tr>
<th>Grading</th>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Final</th>
<th>Lab</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20%</td>
<td></td>
<td>10%</td>
<td>40%</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

Text Book:

Digital Image Processing by Rafael C Gonzalez & Richard E.Woods
Digital Image Processing and Computer Vision by Schalkoff

Reference Books:

Machine Vision and Digital Image Processing Fundamentals by Galbiati
Computer and Robot Vision, by Haralick R.M
Course Name: Cryptographic Techniques
Course Code: COMP 326

Credit Hours: 3
Contact Hours: 2

Level: 7th Level
Prerequisite: 7th Level

Track: □ Core □ Major □ Elective
CS IS CNET

Course Description:
Introduction to design and analysis of cryptographic systems. Symmetric cryptography: Block ciphers and secure hash functions. Asymmetric cryptography: Key exchange and public-key systems. Authentication and encryption in an adversarial model. Simple cryptanalysis. Protocol design and analysis etc.

Course Objectives:
1. The ability to safely integrate cryptographic functionality into computer programs using a cryptographic programming toolkit.
2. The ability to discuss cogently major cryptographic primitives, their design principles, common uses, security goals and methods of analysis.
3. Goals and techniques of cryptanalysis.
4. Programming to avoid security vulnerabilities.
5. Use of reductions to establish security guarantees.

Grading:
- □ Mid-term 20%
- □ Project
- □ Quizzes 10%
- □ Final 40%
- □ Lab 20%
- □ Participation 10%

Text Book:

Reference Books:
Course Name: Genetic Algorithm

Course Code: COMP 444

Credit Hours: 3

Contact Hours: Lec. 2, Lab. 2, Tot. 4

Track: ☑ Core ☐ Major ☑ Elective

Level: 9th Level

Prerequisite: CS, IS

Course Description:
The objective of this course is to prepare the student to use modern design strategies rooted in biological principles. Examples include: the use of biological genetics as an inspiration for genetic algorithms for the automated design of components and systems; the use of linguistic based controls strategies based on human speech and logic to formulate control systems (a strategy formally known as fuzzy logic); the construction of artificial neural networks modelled after the construction of biological nervous systems for information processing and control; use of protein classification and identification algorithms to solve engineering design problems; and the study of immunology and how it relates to computer viruses and synthetic immune systems.

Course Objectives:
The development of complex, highly reliable integrated systems with greatly reduced design cycle times poses a formidable challenge to today's engineers. Traditional engineering design practices are not adequate to meet this new challenge; engineers are increasingly turning to biology for answers. Engineering has always borrowed from nature to provide conceptual examples (for instance, aerodynamics from birds). However, the extraordinary demands placed on today's engineering designs have resulted in the use of biological concepts in a concrete and mathematically defined way. These concepts include artificial networks that mimic the functioning of neurons in the brain, fuzzy logic that more closely reflects human reasoning, and genetic algorithms that imitate the mechanics of biological genetics. Future challenges facing engineering, science, and technology will require multidisciplinary education including proficiency in biologically inspired engineering. The objective of this course is to make students aware of biologically inspired engineering techniques and to equip them with multidisciplinary breadth - thus making them prepared to collaborate with their colleagues from other disciplines.

Grading: ☑ Mid-term 20% ☐ Project ☑ Quizzes 10%

Final 40% ☑ Lab. 20% ☑ Participation 10%

References:

- Koza, J.R. (1994), Genetic Programming II: Automatic Discovery of Reusable Programs, MIT Press
Course Name: Fuzzy Logic  
Course Code: COMP 343  
Credit Hours:  
Contact Hours:  

<table>
<thead>
<tr>
<th>Track</th>
<th>Core</th>
<th>Major</th>
<th>Elective</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS</td>
<td>IS</td>
<td>CNET</td>
<td></td>
</tr>
</tbody>
</table>

Level: 9th Level  
Prerequisite:  

Course Description:
Fuzzy logic is a tool that can be applied to ambiguous, complicated, complex, or nonlinear systems or problems, which cannot easily solved by classical techniques. This course discusses the fundamental of fuzzy set theory and fuzzy logic. In addition, this course also introduces applications of fuzzy logic in several areas such as fuzzy control and fuzzy decision making.

Course Objectives:
In this course you will learn:
(a) how imprecision in concepts can be discussed using the basics of fuzzy sets;
(b) the basic principles of organizing a fuzzy logic system
(c) what is inside the rule-base of a fuzzy control system
(d) about methods of building a fuzzy control system
The course notes comprise Microsoft Excel spreadsheets used for building fuzzy control systems including a fuzzy air-conditioner and how to balance an object on the top of a pole.

Grading

<table>
<thead>
<tr>
<th>Mid-term</th>
<th>Project</th>
<th>Quizzes</th>
<th>Participation</th>
</tr>
</thead>
<tbody>
<tr>
<td>20%</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
</tr>
</tbody>
</table>

Final  
Lab.  
Participation  

Prescribed Text Books
A first course in fuzzy logic
By Hung T. Nguyen, Elbert Walker
Reference Book
Reference Books
Mathematics of Fuzzy logic by Peter Hajek
Course Name: Game Programming
Course Code: COMP 418

Credit Hours: Game Programming
Contact Hours: Lec. 2, Lab. 2, Tot. 4

Track:  ☑ Core  ☑ Major  ☑ Elective
☒ CS  ☑ IS  ☑ CNET

Level: 9th Level
Prerequisite:

Course Description:
This course is designed to introduce the student to the software engineering aspect of game programming. C++ is required. Your class project will be written in C++ with no exceptions. The purpose of this class is to help prepare you for a job as a professional game programmer at a typical game studio; therefore, Java, C#, or any other programming language will not be accepted for your class project. By the end of this class, you should have the basic building blocks of your own personal game engine written from scratch that you can build upon for the purposes of showing off to potential employers once you add your own flair and convert your class project to either your own game or some useful development tool.

Course Objectives:
- Implement large programs using advanced Java features.
- Apply patterns of design and testing to improve program development.
- Utilize trigonometry and vector mathematics to solve game-related problems.
- Apply artificial intelligence techniques to create autonomous agents.
- Use a graphics library to render appropriate visual representations of two-dimensional scenes.

Grading:
- ☑ Mid-term 20%
- ☑ Project
- ☑ Quizzes 10%
- ☑ Final 40%
- ☑ Lab. 20%
- ☑ Participation 10%

Prescribed Text Books:
Programming Game AI by Example by Mat Buckland.