### City University

**Faculty of Science and Engineering**

**Department of Computer Science and Engineering**

**Course Outline**

**Course Code and Course Title:** CS 401, System Programming

**Credit hours:**3.0

**Prerequisites:** Operating System, Computer Architecture

**Semester:** Spring 2020

**Total Weeks:** 12 **Hours/Week:** 3 **Total Hours:**36+

**Course Instructor:** Sharmin Akter

**Designation:** Lecturer

**Phone: +88 01744960557 Email**: [sharmin.cmt0809@gmail.com](mailto:sharmin.cmt0809@gmail.com)

**Class Time: Sunday-Tuesday**

**Class Room: (City Campus)**

### Course Goals:

The goal of this course is to give introductory information of computer system, its organization and its system program. This course will be covered some basic principle of computer system. Today in every works we see the use of computer. We can’t go a single step without computer. Presently we see the application of computer in every spheres of our life. This course will be helpful to design embedded system by using microprocessor and programming.

### Course Description:

The objective of this course is to provide you with a basic understanding of the issues involved in writing system programs on a Linux or Unix system, manipulating system processes, system io, system permissions, files, directories, signals, threads, sockets, terminal, etc. Since the operating system discussed will be Linux, an intermediate knowledge of the Linux Operating System is assumed. In addition, students are expected to have an intermediate knowledge of the C (or C++) programming language and knowledge of the theory of multiprocessing operating systems.

**Intended learning outcomes (ILOs) of the Course**

|  |  |
| --- | --- |
| **KNOWLEDGE** | **LO1 :** will be able to understand the fundamentals of how System works. |
| **LO2** : will be able to apply achieved knowledge |
| **LO3 :** Students will be able to use software quality assurance tools and collaborate them and will be able to solve real life problem. |
| **SKILLS** | Will grow the skill of software system program. |
| Will grow the skill how an existing system work and what are future scope |
| Will be able to share new ideas. |
| **ATTITUDE** | Will develop attitude to group dynamics and team work |
| Will create positive attitude to listen ideas of classmates |
| Will be able to think out of box. |

**Mapping of Course ILO and PLO:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Learning Outcome (LO) of the Course** | **Program Learning Outcome (PLO)** | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **ILO1** | MJ |  |  |  | MN |  |  |  |  | MN | MJ |  |
| **ILO2** | MJ | MJ | MN |  |  |  |  |  |  |  | MJ |  |
| **ILO3** | MJ | MJ |  | MN | MJ |  |  |  | MJ |  | MJ | MN |

**Course Contents**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Topic** | **Teaching Strategy** | **Number of Sessions**  **(Required/Actual)** | **Schedule Date** |
| 1st + 2nd | **Introduction:** Embedding computer, microprocessor, cyber physical system, embedded system design process: requirements, specification, modeling, design, analysis. | Lecture,  Demonstrate analysis. | 2 (2) |  |
| 3rd + 4th | **Embedded processor and CPUs**: Types of processor, microcontroller, DSP processor, x86 architecture, graphics processor, parallel processing, concurrency, pipelining, caching, CISC and RISC processor, multicore processor architecture, supervisor mode, exceptions, traps | Lecture,  Problem Example | 2 (2) |  |
| 5th+6th | **Memory architecture**: Memory device organizations, RAM, ROM, cache, memory hierarchy, memory mappings, | Lecture,Exercise,  Problem Example  Class Test 1 | 2(2) |  |
| 7th+8th | Register file, scratchpad, memory models, memory management and address translator, bus protocol, DMA, AMBA bus. | Lecture,Exercise,  Problem Example  Assignment 1 | 2(1) |  |
| 9th + 10th | **Input Output devices:** I/O hardware, Interrupt, exception, atomicity, analog/digital interface, signal conditioning, sampling, busy-wait I/O, interrupt vector table. | Lecture,  Exercise,  Problem Example | 2(2) |  |
|  | **Midterm Exam** |  |  | |
| 11th + 12th | **Kernel:** General kernel responsibilities, kernel organization, kernel compiling and installing, kernel's role at system startup, process creation and termination, Process execution, | Lecture,  Exercise,  Problem Example | 2 (1) |  |
| 13th +14th | ELF format, inter process communication, signal handling, Memory management: page frame management, memory area management, kernel memory management | Lecture,  Exercise,  Problem Example | 2 |  |
| 15th + 16th | **Linux Module Programming**: linux device driver,Building, installing and loading modules, I/O architecture, the device driver model, device files, character device driver, block device driver, | Lecture, Exercise,  Problem Solving | 2 |  |
| 17th + 18th | **Assembler, Linker & Loaders:** Basic Assembler Functions, Machine Dependent Assembler features, Machine Independent Assembler Features | Lecture, Exercise,  Problem Solving  Class Test 2 | 2 |  |
| 19th + 20th | Assembler Design Options - One pass assembler and multipass assembler, Basic Loader Functions, | Lecture, Example  Assignment 2 | 2 |  |
| 21st + 22nd | Machine Dependent Loader Features, Machine Independent Loader Features, Linkage Editors, | Lecture, Example | 2 |  |
| 23rd | **Program design and analysis:** State machines, control/data flow graph, assemblers, linker, program optimization, program level performance analysis, software performance optimization, clear-box testing, black-box testing. | Lecture, Example  Presentation | 2 |  |
|  | **Final Exam** |  |  |  |
| **Total** | | | **23** |

**Teaching & Learning Method**

|  |
| --- |
| Use of electronic presentation with multimedia in class. |
| Team work. |
| Problem analysis and discussion. |
| The course consists of lectures, tutorials, and laboratory sessions. You will need to work outside of these formal hours to complete your project and assignments and to study and reflect upon the course material |

**Assessment Schedule**

|  |  |  |
| --- | --- | --- |
| Assessment 1 | Quizzes | Week 4, Week 10 |
| Assessment 2 | Assignments | Week 5, Week 11 |
| Assessment 3 | Presentation | Week 5, Week 11 |
| Assessment 4 | Mid-Term Exam | Week 7 |
| Assessment 5 | Final Exam | Week 13 |

**Weights of Assessments**

|  |  |
| --- | --- |
| Assessments | % |
| Mid-Term Exam | 30 |
| Final Exam | 40 |
| Quizzes | 10 |
| Assignments | 10 |
| Presentation | 10 |
| **Total** | **100** |

**List of References**

**Text Book**

1. Computer as Components – Wayne Wolf.  
2. Unix System Programming – Terrence Chan.