National University of Computer and Emerging Sciences

FAST School of Computing **Fall 2019** **Islamabad Campus**

CS-220: Operating Systems

Serial No:

Sample Exam

Total Time: 90mins

Total Marks: 50

Monday, 24th May, 2021

Course Instructors

Faisal Cheema, Sidra Khalid, Shoaib Mehboob, Hasan Mujtaba

_____

Signature of Invigilator

_____ _____

Student Name Roll No Section Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. Attempt on question paper. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
2. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
3. If you need more space write on the back side of the paper and clearly mark question and part number etc.
4. After asked to commence the exam, please verify that you have three? (3?) different printed pages including this title page. There are a total of 5? questions.
5. Calculator sharing is strictly prohibited.
6. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.

	Q-1	Q-2	Q-3	Total
Marks Obtained				
Total Marks	**20**	**20**	**10**	**50**

Q1: Consider the following situation of the Ready Queue [20 Marks]

P Name	P ID	Arrival Time	CPU Burst	Priority
system	210	0	10	5
chrome	138	2	5	6
interrupts	521	1	15	7
explorer	987	5	10	3
desktop	333	3	20	2

Assuming all times mentioned are in seconds; calculate the **average waiting time** with the following algorithms:

Note: You must show clear, complete and correct Gantt charts for each algorithm in the **space provided below**. Make sure to complete your calculations and write the answer clearly in the column to the right. If you fail to comply with these guidelines you will be awarded zero marks. Sign at the end of this line for one bonus mark.

Approach	Average Waiting Time
Shortest Job First:	
First Come First Serve:	
Preemptive Priority based Scheduling:	
Round Robin (Time Quantum of 3s)	

Q2a.

How many child processes will be created through the code given below? Draw the complete process tree. [5 Marks]

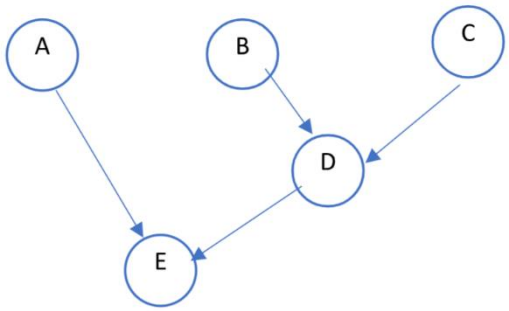
Program

```
#include<iostream>
#include<unistd.h>
using namespace std;

int main(){

   if( !fork() && fork() ){

      if( fork() || !fork() ){

         cout<<" * ";

      }

   }

}
```

Q2b.

Based on fork(), wait() and exec; create a custom shell for the Linux Operating Systems. [15 Marks]

Q3. We have five functions that solve a single problem together. Suppose these functions, labelled A through E, depend on each other according to the following graph. [10 marks]



A B C D E Each edge of the graph denotes a dependency between two of these functions. For example, the edge from node B to node D means that functionB must be called, and must return, before functionD can be called.

Write a Pthreads program to execute the above five functions in a way that is maximally parallel (i.e., always runs as many threads in parallel as possible), adheres to the above dependency graph, and uses the minimal number of threads possible (including the main() thread). Your solution should still use only pthread_create for thread creation and pthread_ join() for synchronization. You need to use below Methods for worker threads are given below:

```
void *threadA(void *vargp){ functionA(); }

void *threadB(void *vargp){ functionB(); }

void *threadC(void *vargp){ functionC(); }

void *threadD(void *vargp){ functionD(); }

void *threadE(void *vargp){ functionE(); }
```