National University of Computer and Emerging Sciences, Lahore Campus

	Course:	Operating System	Course Code:	CS-205
	Program:	BS(Computer Science)	Semester:	Fall 2018
	Duration:	1 hour	Total Marks:	40
	Paper Date:	2nd October, 2018	Weight:	15%
	Section:	All	Page(s):	4
	Exam:	Mid-1	Roll No.	

Instructions/Notes: Answer questions on the question paper. Write answers clearly and precisely, if the answers are not easily readable then it will result in deduction of marks. Use extra sheet for rough work, **cutting and blotting on this sheet will result in deduction of marks**.

Question 1 (3 points): List four main components of computer system which the kernel has to manage.

(1)

(2)

(3)

(4) I/O including network management

Question 2 (2 points): The CPU is connected to a _____(a)_____, which is connected to the device controller, which is connected to a _____(b)_____ device. This is the path of communication that enables the hardware interrupts to occur.

(a)

(b)

Question 3 (2 points): Which of the following scheduling algorithms is non-preemptive?

(a) Round Robin

(b) FCFS

(c) Shortest Remaining Time First

Question 4 (2 points): There are machine level commands written inside the CPU which, when executed, can change the values of the registers built inside the device controllers.

(a) True

(b) False

Question 5 (5 points): Write in each cell what type of Inter Process Communication is discussed. Tick the correct column.

Scenario	Shared Memory	Message Passing
Done correctly, sharing of information is faster using this technique		
Processes use `write()` and `read()` system calls		
In some forms only one way communication is possible at one time		
The processes must use some synchronization mechanism		
A queue is used and mostly the queue is controlled by the kernel		

Question 6 (2 points): Suppose a machine runs one instruction in one clock cycle. Now suppose a **program** has 10 instructions in its instruction stream. The program is loaded into memory and becomes a "process". There is no way that the process takes more than 10 clock ticks to finish, is this correct?

(a) Yes

(b) No

Question 7 (4 points): Name any two methods used for parameter passing between a process and the kernel.

(a)

(b)

Question 8 (5 points): Tell the output of the following code. Assume that the parent process running following code has the $PID = 100$. Each new `fork()` creates a new process. Each child process gets the process ID in following way. The first digits of the child process ID are all borrowed from the parent. The last digit is equal to the number of `fork()` done by the parent. For example, if parent whose $PID = 100$, the child created in result of the first fork will have the $PID = 1001$ and the child created in result of the second fork will have $PID = 1002$.
Assume that each instruction runs in the order. Meaning instructions written on smaller line numbers will necessarily execute before the instructions written on bigger line numbers.
Hint:The function `getpid()` returns the *PID* of the calling process.

```
1   # include <stdio.h>
2   int main(void)
3   {
4   int pid=0;
5   pid = fork();
6   if ( pid == 0)
7   {
8           printf("%d,", getpid());
9           pid = fork();
10          if ( pid == 0)
11          {
12                  printf("%d,", getpid());
13                  pid = fork();
14                  if ( pid > 0)
15                  {
16                          printf("%d,", getpid());
17                          pid = fork();
18                          if ( pid > 0)
19                          {
20                                  printf("%d,", getpid());
21                                  pid = fork();
22                                  if ( pid == 0)
23                                  {
24                                          printf("%d,", getpid());
25                                  }
26                          }
27                  }
28          }
29
30  }
31  else if (pid > 0)
32  {
33          printf("%d,", getpid());
34  }
35  return 0;
36  }
```

Question 9 (5 points): Inspired from the above code, write a similar code using `fork()` which prints the string "100,1001,1002,1003" (without the quotes). You can make the same assumption about the execution order as above. Meaning the line written before will execute before.

Question 10 (10 points): Suppose in a machine the CPU executes one instruction per clock cycle. There are three Ethernet cards in the machine. Each machine runs some CPU cycle then reads data from any of the Ethernet cards. The processes arrive in order, i.e. P_1 then P_2 and then P_3

Explanation: The columns CPU Burst+Length show the number of CPU clock cycles needed by the process. The columns I/O Burst+Length show the number of Ethernet cycles needed after each CPU Burst. The table only shows the order in which processes need those cycles, how they will execute depends upon the scheduling algorithm.

- Using the FCFS algorithm list down the order of execution of the processes.
- Calculate the total time of execution of all processes.
- Calculate the average waiting time.

Process Name	Length	CPU Burst	I/O Burst
P_1	3	Yes	-
P_2	6	Yes	-
P_3	8	Yes	-
P_1	12	-	Yes
P_2	4	-	Yes
P_3	7	-	Yes
P_1	7	Yes	-
P_2	5	Yes	-
P_3	3	Yes	-
P_1	13	-	Yes
P_2	10	-	Yes
P_3	7	-	Yes
P_1	3	Yes	-
P_2	25	Yes	-
P_3	12	Yes	-
P_1	17	-	Yes
P_2	15	-	Yes
P_3	8	-	Yes
P_1	3	Yes	-
P_2	3	Yes	-
P_3	3	Yes	-