Reg. No.:
Name :

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Continuous Assessment Test-1 – May 2023

Programme	:	B.Tech (CSE and its Specialization)	Semester	:	Fall Inter 2022-23
Course	:	Operating System	Code	:	BCSE303L
Faculty	:	Dr.ABDUL QUADIR MD Dr.BHANU CHANDER BALUSA Dr.RISHIKESHAN C A Dr.SANDEEP KUMAR SATAPATHY Dr.PRADEEP K Dr.VALARMATHI Dr.INDRA PRIYADHARSHINI Dr.MONICA K M Dr.SANGEETHA N Dr.ANANDAN P Dr.YOGESH C Dr.LEKI CHOM THUNGON Dr.TAPABRATA ROY	Slot(s)	:	E2+TE2
			Class Nbr(s)	:	CH2022232500759 CH2022232500982 CH2022232500983 CH2022232500993 CH2022232500992 CH2022232500988 CH2022232500755 CH2022232500756 CH2022232500757 CH2022232500758 CH2022232500760 CH2022232501076 CH2022232501075
Time	:	1½ Hours	Max. Marks	:	50

Answer **ALL** Questions

1.		Mr. Anuj is attending an interview for a developer role at XYZ Pvt. Ltd. During the interview his manager asked him to list out the components and functionalities of the Operating system. Your task is to help Mr. Anuj to achieve the above scenario while taking into consideration the goal of a good operating system?	[10]
2.	a.	Consider the following C program to be executed on CPU. With neat diagram explain how the following code is prepared for execution in terms of text, data, stack and heap segments when stored in memory. **[5 Marks]**	[10]

```
#include<stdio.h>
int c;
int main()
{
int a=10;
int b=20;
print(a,b);
return 0;
}
void print(int x, int y)
{
  c=x+y;
  printf("%d",c);}
```

	b.	In general, the processor does not acknowledge the interrupts generated, till the completion of current instruction. Assume that the processor acknowledges the generated interrupt by preventing the execution of current on-going instruction. Write the various difficulties encountered by the operating system if the above step is implemented by the processor. **[5 Marks]**	

3. a. Consider the following C program and explain in detail the creation of child processes and how many times the "Hello" statement will be printed and which process will be responsible for printing each of them. **[5 Marks]** **[10]**

```
#include<stdio.h>
#include<unistd.h>
int main()
{
        if (fork() && fork())
        {
                fork();
        }
        printf("Hello\n");
        return 0;
}
```

b. Find the output for the two programs given below. Compare in brief why there is a similarity or difference in the outputs. **[5 Marks]**

```
#include <stdio.h>
#include<unistd.h>
int main()
{
   int i, a = 10;
   for(i = 0; i < 2; i++)
        {
      a += 5;
      printf("%d\n", a);
        }
   return 0;
}
```

```
#include <stdio.h>
#include<unistd.h>
int main()
{
   int a = 10;
   fork();
   a += 5;
   printf("%d\n", a);
   return 0;
}
```

4. Assume four students [S1, S2, S3, S4] would like to refer operating system book in the VIT library. Unfortunately, only one copy is available for reference. Assuming S1 will read the book for 10m, S2 for 12m, S3 for 6m, and S4 for 9m. Also consider that all four students arrive at the library at the same time. Analyze the following scenarios with appropriate CPU scheduling algorithm and determine the average Turn Around Time (TAT) and average Waiting Time(WT) of the students in library if, **[10]**

   i. The students are allowed to read the book based on their time requirement. The book will be allotted (until completion) to a student only if the demanded reading time of the student is lesser than the others.

   ii. The students are allowed to read the book one after the other for 5m in each of their turn.

5. a. Consider the following snapshot of a system in which five resources R1, R2, R3, R4 and R5 are available. Using the below snapshot convert the matrix representation to a resource **[10]**

allocation graph and find whether the system contains a deadlock or not. **[5 Marks]**

	Allocation					*Request*					*Available*				
	R_1	R_2	R_3	R_4	R_5	R_1	R_2	R_3	R_4	R_5	R_1	R_2	R_3	R_4	R_5
P_1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
P_2	1	0	0	0	0	0	1	1	0	1					
P_3	0	0	1	0	0	0	0	0	0	1					
P_4	0	1	0	0	0	0	0	0	0	0					
P_5	0	0	0	0	1	0	0	0	1	0					

6. Consider the following snapshot of a system in which five resources A, B, C, D and E are available. Using the below snapshot convert the matrix representation to a resource allocation graph and find whether the system contains a deadlock or not **[5 Marks]**

	Allocation					*Request*					*Available*				
	A	*B*	*C*	*D*	*E*	*A*	*B*	*C*	*D*	*E*	*A*	*B*	*C*	*D*	*E*
P_0	1	0	1	1	0	0	1	0	0	1	2	1	1	2	1
P_1	1	1	0	0	0	0	0	1	0	1					
P_2	0	0	0	1	0	0	0	0	0	1					
P_3	0	0	0	0	0	1	0	1	0	1					

□□□