Name Surname   : ....

Signature:.........................

Student Id   : ....

Q 6. (6p) Draw a directed graph representing the process state model. Name all edges and vertices.

Q 7. (10p) Considering the following code (Assume that fork does not fail.):

c. Draw the process tree shown parent (P_{id}) and child (C_{id}) processes where n is the ID of the process.

```
int main () {
    fork();
    if(fork())
        printf(" A ");
    else
        fork();
    printf(" B ");
    wait(NULL);
    return();
}
```

a. How many times "A" is printed on the screen.

....

b. How many times "B" is printed on the screen.

....

Q 8. (9p) What could the output of the **concurrent** execution of process A and process B be? (State all possible outputs)

Initialization of shared variables
int x=2;
int y=0;

Process A	Process B
while(x==2) {do-nothing};	printf("L");
printf("E");	x=1;
y=1;	while(y==0){do-nothing};
y=0;	printf("A");
printf("M");	
y=1;	

....

....

Q 9. (10p) Consider P_1 and P_2 processes that require T_1 to happen before T_2. Show a semaphore based solution to this problem using the semaphore variable S.

....

Akdeniz University, Engineering Faculty
Computer Engineering Department
CSE 303 Fundamentals of Operating Systems Midterm Exam

Q 1. (8p) Define the following terms and give an example;
a) Operating System:

....

....

Example:..

b) Pipeline:

....

....

Example:..

Q 2. (13×2p) Mark the following (T-F) questions using T for True, and F for False..
(......) Q2-1 A Multi-threaded process has two program counters per thread.
(......) Q2-2 The short-term scheduler controls the degree of multiprogramming.
(......) Q2-3 Interrupt-driven I/O provides better performance when moving large amounts of data than DMA.
(......) Q2-4 $@ symbol in a Makefile represents the left side of the : symbol.
(......) Q2-5 A child process can only be an orphan process while its parent can be either orphan or a zombie process.
(......) Q2-6 There must be a space character in the beginning of any command in a Makefile.
(......) Q2-7 When using fork system call parent and child process have the same address space.
(......) Q2-8 With NUMA, some parts of memory may take longer to access than other parts.
(......) Q2-9 CD-R and DVD-R are examples for WORM devices.
(......) Q2-10 CPU registers has faster access time than any other device including CPU cache memory.
(......) Q2-11 Privilege escalation allows user to change file access permissions.
(......) Q2-12 Emulation used when source CPU type different from target type.
(......) Q2-13 Operations in Message Passing architecture is faster than shared memory architectures.

Q 3. (12×2p) Fill in the blanks with appropriate terms.
a) A .... or .... is a software-generated interrupt caused either by an error or a user request.
b) When an interrupt occurs, the operating system preserves the state of the CPU by storing every ....
and the ....
c) The .... instruction moves a byte or word from main memory to an internal register within the CPU
while the .... instruction moves the content of a register to main memory.
d) The sequence of steps that the CPU follows to process instructions is called as .... Cycle.
e) In a multiprocessor environment all CPUs must have the most recent value in their cache which is known
as ....
f) In the context of Cloud Computing, Google Docs is an example for ....
g) One of the example Shell program in a Linux/UNIX systems is ....
h) One method for system call parameter passing is to use ....
i) PID value of 1 is assigned to the .... process on Linux Systems.

Q 4. (3p) Explain the difference between the core dump and crush dump.

....

....

Q 5. (4p) Write the manual compilation and linking steps to compile a project having main.c, plib.c, and plib.h files into
main.exe using gcc compiler.

....

....

5

Threads share the memory and the resources of the process by default
(4 Puan)

→ ○ True

○ False

6

Binary semaphore equals to mutex variables.
(3 Puan)

→ ○ True

○ False
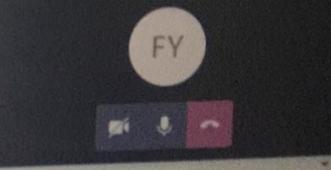
7

In Direct Memory Access, one interrupt is generated per byte
(4 Puan)

○ True

○ False

○ Meeting in "Week 10 ...   13:24   —

FY

09:32
11.12.2020

2

Explain the DIFFERENCE between the core dump and crash dump.
(3 Puan)

Yanıtınızı girin

3

When using fork system call parent and child process have the same address space.
(3 Puan)

True

False

4

Program code is stored in the text section of the program
(3 Puan)

True

False

1

Which parts of the program is shared among the threads. 🔊
(4 Puan)

○ Code section, data section, open files

○ Code section, registers, open files

○ Data section, stack, registers

○ program counter, registers, stack

12

The heap contains dynamically allocated memory during run time
(3 Puan)

➡ True

○ False

13

A trap or exception is a software-generated interrupt caused either by an error or a user request
(3 Puan)

➡ True

○ False
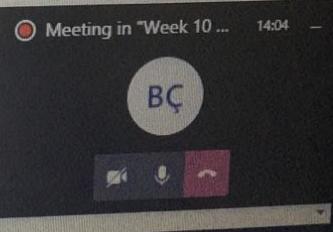
Growing in popularity as numbers of threads increase, program correctness more difficult with explicit threads

14

Give an example method for implicit threading?
(3 Puan)

OpenMP, Thread Pools, Grand Central Dispatch

Yanıtınızı girin

15

Operations in Message Passing architecture is faster than shared memory architectures.
(4 Puan)

○ True

➡ False

16

The size of a character pointer equals the size of a double-pointer in C programming.

➡ True

○ False

17

Ordinary pipes need a parent-child relationship.
(3 Puan)

➡ True

○ False

8

CPU registers have a faster access time than any other device including CPU cache memory.
(2 Puan)

➡ True

⭕ False

9

What are the main sections in the critical section problem? 🗊
(4 Puan)

Y entry section, critical section, exit section, remainder section

10

In the case of concurrency, the execution of the threads will be interleaved over time
(3 Puan)

➡ True

18

Write the pseudocode for Peterson's Solution for the critical section problem
(5 Puan)

Yanıtınızı girin

```
do {
flag[i] = true;
turn = j;
while (flag[j] && turn = = j);
critical section
flag[i] = false;
remainder section
} while (true);
```

19

Synchronous threading does not involve data sharing among threads.
(2 Puan)

○ True

⇨ False

Parent thread creates one or more children and then must wait for all
of its children to terminate before it resumes
Known as fork-join strategy
Threads can run concurrently but parent cannot continue until this
work has been completed

20

Function parameters, return addresses, and local variables stored in a Hash data structure for
processes
(3 Puan)

STACK

○ True

3

Considering the following code segment
a) Draw the process tree. Each node should present the up to date values of the variables a,b and i
b) Provide the unordered list of print outs.

You should upload your answer as a jpg file. Before submitting, be sure that writings and drawings are clearly presented. (Anonim olmayan soru ⓘ)
(10 Puan)

4

If 20% of an application can be executed in parallel, what will be the theoretical maximum speedup?
(3 Puan)

Yanıtınızı girin

5

7

Process	Burst Time
P_1	10
P_2	29
P_3	3
P_4	7
P_5	12

Considering the given processes and their CPU burst times, compare the FCFS and RR scheduling algorithms using the average waiting time metric. Quanta q=5.

(Anonim olmayan soru ⓘ)

(20 Puan)

↑ Dosyayı karşıya yükle

Dosya sayısı üst sınırı: 1   Tek dosya boyutu üst sınırı: 10MB   İzin verilen dosya türleri: Word, Excel, PPT, PDF, Resim, Video, Ses

	Allocation					Max					Available				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
P0	2	0	0	1	0	4	2	1	2	3	3	3	3	1	2
P1	0	1	2	1	1	3	2	5	2	3					
P2	2	1	0	3	1	2	3	1	7	3					
P3	1	3	1	0	2	1	4	2	4	4					
P4	1	4	3	2	2	3	6	6	5	4					

Consider the given snapshot of a system: Answer the following questions using the banker's algorithm:
a) How many resources are there for each resource type in this system?
b) Is this system safe? If yes provide a safe sequence. If not explain why?
c) If a request from P4 arrives for (0,0,2,0,1) can the request be granted immediately? Explain why? (Anonim olmayan soru ⓘ)
(20 Puan)

↑ Dosyayı karşıya yükle

Dosya sayısı üst sınırı: 1   Tek dosya boyutu üst sınırı: 10MB   İzin verilen dosya türleri: Word, Excel, PPT, PDF, Resim, Video, Ses

5

A doubling scheduler uses a prioritized round-robin scheduling policy. New processes are assigned an initial quantum of length R. Whenever a process uses its entire quantum without blocking, its new quantum is set to twice its current quantum (2*R). If a process blocks before its quantum expires, its new quantum is reset to R. For the purposes of this question, assume that every process requires a finite total amount of CPU time. If the scheduler gives higher priority to processes that have smaller quanta. Is starvation possible in this system? Explain briefly.

(10 Puan)

Yanıtınızı girin

4

Select the correct option for the outputs of the concurrent execution of process A and process B to be
(15 Puan)

Initialization	
int x=2; int y=3;	
Process A	**Process B**

`while(x==2) {do-nothing};`
 `printf("S");`
 `y=x-y;`
 `printf("3");`
 `y=1;`
	`printf("C");`
	 `x=y*x;`
	 `printf("E");`
	 `while(y==3){do-nothing};`
	 `printf("0");`
	 `x=x-y;`
	 `printf("3");`

I. CS3E03   II. CES303   III. CSE303   IV. CE03S3 V. S3CE03

○ I, III, and IV

○ III and IV

○ III

⦿ I, II, and III

○ All of them

3

If the semaphore operations Wait and Signal are not executed atomically, then mutual exclusion may be violated. Assume that Wait and Signal are implemented as below: Lx represents the line numbers.

L1 void Wait (Semaphore S) {
L2   while (S.count <= 0) {;}
L3   S.count = S.count - 1;
L4 }
L5 void Signal (Semaphore S) {
L6   S.count = S.count + 1;
L7 }

Describe a scenario of context switches (CS) where two threads, T1 and T2, can both enter a critical section guarded by a single mutex semaphore as a result of a lack of atomicity. In your scenario clearly describe the initial value of the semaphore S, and sequence of operations using the line numbers Lx. For example;
S=5;T1-L1:L3,CS,T2-L5,CS,T1-L4,CS,T2-L6:L7 means that Semaphore is initialized with value of 5. Then T1 executes lines from 1 to 3. A context switch occurs, then T2 executes line 5. Another context switch occurs

(10 Puan)

Yanıtınızı girin

1

Writer Process	Reader Process

	wait(mutex);
	    readcount = readcount + 1;
	    if *readcount = 1* **then** *wait(wrt);*
	signal(mutex);

wait(wrt);
. . .
    writing is performed
. . .
signal(wrt);
	. . .
	    reading is performed
	. . .
	wait(mutex);
	    readcount := readcount - 1;
	    if *readcount = 0* **then** *signal(wrt);*
	signal(mutex);

Considering the readers and writers problem, assume that rules have changed. The rule "Multiple readers can read at the same time" is now changed to "Up to 5 readers can read at the same time". Here is the pseudocode for the original readers-writers problem. Rewrite the code to meet the conditions for the new rule.

Rules:
1. Only one writer writes at a time
2. While writing reading is not allowed
3. While reading writing is not allowed
4. (NEW) Up to 5 readers can read at the same time (Anonim olmayan soru ⓘ)

(15 Puan)

```
...
int main()
{
pid_t smith;
int a=2; int b=3;
smith = fork( );
if (smith == 0) {
    fork( );
    a++;
    fork( );  /* BEWARE */
}
else if (smith > 0) {
    b++;
    fork( );      /* BEWARE */
}
printf("%d %d",a,b);
}
```

Here is the code for a program named Agent_Smith.c. Including the initial parent process,
A)   How many Agent_Smith processes are created? Assume there are no errors.
B)   Draw the process tree showing the up-to-date variables a and b. (Anonim olmayan soru ⓘ)

(10 Puan)

⬆ Dosyayı karşıya yükle

Dosya sayısı üst sınırı: 1   Tek dosya boyutu üst sınırı: 10MB   İzin verilen dosya türleri: Word, Excel, PPT, PDF, Resim, Video, Ses

...... Engineering Department

Q 1. (8p) Define the following terms and give an example;
a) Operating System:

....

....

....

Example:..
b) Pipeline:

....

....

....

Example:..

Q 2. (13×2p) Mark the following (T-F) questions using T for True, and F for False..
(.....) Q2-1 A Multi-threaded process has two program counters per thread.
(.....) Q2-2 The short-term scheduler controls the degree of multiprogramming.
(.....) Q2-3 Interrupt-driven I/O provides better performance when moving large amounts of data than DMA.
(.....) Q2-4 $@ symbol in a Makefile represents the left side of the : symbol.
(.....) Q2-5 A child process can only be an orphan process while its parent can be either orphan or a zombie process.
(.....) Q2-6 There must be a space character in the beginning of any command in a Makefile.
(.....) Q2-7 When using fork system call parent and child process have the same address space.
(.....) Q2-8 With NUMA, some parts of memory may take longer to access than other parts.
(.....) Q2-9 CD-R and DVD-R are examples for WORM devices.
(.....) Q2-10 CPU registers has faster access time than any other device including CPU cache memory.
(.....) Q2-11 Privilege escalation allows user to change file access permissions.
(.....) Q2-12 Emulation used when source CPU type different from target type.
(.....) Q2-13 Operations in Message Passing architecture is faster than shared memory architectures.

Q 3. (12×2p) Fill in the blanks with appropriate terms.
a) A .... or .... is a software-generated interrupt caused either by an error or a user request.
b) When an interrupt occurs, the operating system preserves the state of the CPU by storing every ....
and the ....
c) The .... instruction moves a byte or word from main memory to an internal register within the CPU
while the .... instruction moves the content of a register to main memory.
d) The sequence of steps that the CPU follows to process instructions is called as .... Cycle.
e) In a multiprocessor environment all CPUs must have the most recent value in their cache which is known
as ....
f) In the context of Cloud Computing, Google Docs is an example for ....
g) One of the example Shell program in a Linux/UNIX systems is ....
h) One method for system call parameter passing is to use ....
i) PID value of 1 is assigned to the .... process on Linux Systems.

Q 4. (3p) Explain the difference between the core dump and crush dump.

....

....

Q 5. (4p) Write the manual compilation and linking steps to compile a project having main.c, plib.c, and plib.h files into
main.exe using gcc compiler.

....

....

Q 6. (6p) Draw a directed graph representing the process state model. Name all edges and vertices.

Q 7. (10p) Considering the following code (Assume that fork does not fail.):

c. Draw the process tree shown parent (P_{id}) and child (C_{id}) processes where n is the ID of the process.

```c
int main(){
    fork();
    if(fork())
        printf(" A ");
    else
        fork();
    printf(" B ");
    wait(NULL);
    return();
}
```

a. How many times "A" is printed on the screen.

....

b. How many times "B" is printed on the screen.

....

Q 8. (9p) What could the output of the **concurrent** execution of process A and process B be? (**State all possible outputs**)

Initialization of shared variables	
int x=2;	
int y=0;	
Process A	**Process B**
while(x==2) {do-nothing};	printf("L");
printf("E");	x=1;
y=1;	while(y==0){do-nothing};
y=0;	printf("A");
printf("M");	
y=1;	

....

....

....

Q 9. (10p) Consider P_1 and P_2 processes that require T_1 to happen before T_2. Show a semaphore based solution to this problem using the semaphore variable S.

....

....

ne Surname   : ....

lent Id   : ....

Signature:.......................

03/01/2020 11:
Duration: 90 m
Fall 2019 - CSE 303 Fin

8. (6p) In lecture we said that, if the semaphore operations Wait and Signal are **not executed atomically**, then mutua
exclusion may be violated. Assume that Wait and Signal are implemented as below: **Lx** represents the line numbers

```
L1 void Wait (Semaphore S) {
L2     while (S.count <= 0) {;}
L3     S.count = S.count - 1;
L4 }

L5 void Signal (Semaphore S) {
L6     S.count = S.count + 1;
L7 }
```

Provide a scenario of **context switches (CS)** where two threads, **T1** and **T2**, can both enter a critical section guarded
by a single mutex semaphore as a result of a lack of atomicity. In your scenario clearly describe the initial value of
the semaphore S, and sequence of operations using the line numbers **Lx**. For example;
S=5;T1-L1:L3,CS,T2-L5,CS,T1-L4,CS,T2-L6:L7 means that Semaphore is initialized with value of 5. Then T1
executes lines from 1 to 3. A context switch occurs, then T2 executes line 5. Another context switch occurs. etc....

....

9. (10p) Consider a pizza factory in Yakut Bazaar that has 10 pizza plates. The producer is the delivery person and
the consumer is the student. Initially the factory has 3 pizza ready to serve. We use the following three semaphores:
```
semaphore mutex;
semaphore fullBuffer /* Number of filled plates */
semaphore emptyBuffer /* Number of empty plates */
```
Given functions (see code) delivery_person() and student():
a) What will be the initial values of the semaphores?
b) Write a solution that guarantees mutual exclusion and no deadlocks. (Part of the code for student () functi
has been written)

```
/* Initialize */
#define NUM_PLATES    .............
semaphore mutex       = .............
semaphore fullBuffer  = .............
semaphore emptyBuffer = .............
delivery_person() {
.................................................
.................................................
.................................................
.................................................
.................................................
put_a_pizza();
.................................................
.................................................
.................................................
.................................................
.................................................
}
```

←———— Write correct numbers here

```
student() {
.................................................
.................................................
.................................................
.................................................
.................................................
wait(fullBuffer);
wait(mutex);
take_a_pizza ();
.................................................
.................................................
.................................................
}
```

Bonus +5p) What is the latest official versions of the following software
  a. Linux kernel version : ....
  b. Android OS version : ....
  c. Windows OS version : ....
  d. gcc version : ....
  e. JDK version : ....

Q 3. (12p) A scheduler uses a prioritized round-robin scheduling policy. New processes are assigned an initial quantum of length q. Whenever a process uses its entire quantum without blocking, its new quantum is set to twice its current quantum. If a process blocks before its quantum expires, its new quantum is reset to q. For the purposes of this question, assume that every process requires a finite total amount of CPU time.

(a) Suppose the scheduler gives higher priority to processes that have larger quanta. Is starvation possible in this system, i.e., is it possible that a process will never complete because it is neglected by the scheduler? Explain briefly.

_____

_____

_____

_____

_____

(b) Suppose instead that the scheduler gives higher priority to processes that have smaller quanta. Is starvation possible in this system? Explain briefly.

_____

_____

_____

_____

Q 4. (15p) Considering the following C code;
  a. Draw the process tree showing the up to date variable values.
  b. What would be a possible output?
  c. How many lines are printed on the screen (Assume that fork does not fail.)

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(void)
{
    int a,b;
    a=1;
    b=a+1;
    while(a<4){
        if(fork())
            b++;
        else
            a++;
        a+=1;
    }
    printf("%d %d\n",a,b);
    return(0);
}
```

b) 3,2

c) 2,3
   3,4

1. (6×2p) Mark the following (T-F) questions using T for True, and F for False into the given table above.

Q1-1 (.T.) In a system with only one instance of each resource type, the presence of a cycle in the wait for graph sufficient to detect a deadlock.

Q1-2 (.F..) There are solutions to synchronization problems that can be implemented using mutexes, but cannot implemented using semaphores.

Q1-3 (.T.) The convoy effect is a result of using SJF scheduling algorithm.

Q1-4 (..F.) FCFS is a non-preemptive scheduling algorithm.

Q1-5 (.T.) In multithreaded programming each thread has its own registers, stack and code segment.

Q1-6 (.T..) A very large quantum (q) value in RR scheduling is equivalent to FIFO.

2. (20p) An operating system uses a preemptive scheduler algorithm. For the values given table below:

Proc. ID	Submission Time	Required Computing Time
A	0	4
B	1	6
C	3	3
D	4	1

(a) Assuming a scheduling quantum of 2 time units and the running process is preempted only if its quantum expires or it receives its required computing time and exits. What is the **response time** and **turnaround ti** for each process?

(b) Assuming that the scheduler is non-preemptive and uses the shortest-job-first (SJF) heuristic, What **response time** and **turnaround time** for each process?