Test 2 W24 - Results

Attempt 1 of 1

Written Mar 23, 2024 2:09 PM - Mar 23, 2024 2:47 PM

Released Apr 2, 2024 4:00 PM

Attempt Score **83.13 %**

Overall Grade (Highest Attempt) **83.13 %**

Question 1 — 0 / 1 point

A memory fault is more correctly characterized as _____.

- a mode switch
- ✗ a trap
- ➡ an ordinary interrupt
- a supervisor call

Question 2 — 1 / 1 point

Threads may be used in a single-user multiprocessing system for which of the following. (select all that apply)

- ✓ asynchronous processing
- ✓ foreground and background work
- ✓ cache write operations
- ✓ multiple interrupts

Question 3 — 1 / 1 point

A Linux process is in the _____ state if it has been terminated but, for some reason, still must have its task structure in the process table.

- Choose this if none of the others is correct
- ✔ zombie
- interruptible
- suspended
- stopped

Question 4 **0.75 / 1 point**

Which are advantages of the pure KLT approach over the pure ULT approach? (select all that apply)

- ➡ ✔ If one thread is blocked, the kernel can schedule another thread from the same process.
- ✔ Transfer of control from one thread to another within the same process requires a mode switch to the kernel.
- ➡ ✔ Kernel routines themselves can be multithreaded.
- ➡ ✘ The kernel can simultaneously schedule multiple threads from the same process on multiple processors.

Question 5 **0.5 / 1 point**

When the theoretical speedup predicted by Amdahl's Law is not achieved in practice, this is *typically* due to the following. (select all that apply)

- ➡ ✘ overhead due to cache coherence
- ✘ overhead due to calculating the amount of serial code in the application
- ✔ a majority system's processors are malfunctioning
- ➡ ✔ overhead of distributing work to multiple processors

Question 6 **1 / 1 point**

Suppose the following statements comprise the body of a C program.

When it is run, how many X are printed, assuming calls don't fail? Enter a single integer only. No spaces, no decimal, etc.

```
fork();
if ( (int) fork() != 0)
      fork();
printf ("X");
return (0);
```

Answer:

6 ✓

Question 7 **1 / 1 point**

The collection of program, data, stack, and attributes is typically referred to as the _____ .

- process structure
- process location
- Choose this if none of the others is correct
- ✓ process image
- process control block

Question 8 **1 / 1 point**

Relevant parts of a program are given below. Assume the program compiles, runs and prints 2 lines. Assume file "myfile" contains the character whose ascii code is 80. What is most likely the *first* line printed by the program? Enter a single integer only (no decimal, spaces etc.)

```
int X, fd;
int Func() {
   X=70;
   close(fd);
   _exit(0);
}
int main(void) {
   // st is allocated here
   X=43;
   fd=open("myfile",O_RDONLY);
```

```
    clone(Func, st, 0, NULL);
    sleep(1) //give child time to complete
    printf("%d\n",X);
    if (read(fd, &tempch, 1) < 1) printf("0\n"); //zero
    else printf("%d\n", tempch);
    return 0;
}
```

Answer:

43 ✔

Question 9 **1 / 1 point**

Which are among the key states for a thread? (select all that apply)

- ✔ ☐ Blocked
- ✔ ☐ Block-Suspend
- ✔ ☐ Ready-Suspend
- ✔ ☐ Running

Question 10 **0.75 / 1 point**

When these occur, control is given to the OS (select all that apply):

- ✔ ☐ a cache read
- ➡ ✔ ☐ a trap
- ➡ ✔ ☐ a system call
- ➡ ✖ ☐ an interrupt

Question 11 **1 / 1 point**

Suppose an interrupt occurs during execution of process P. If P continues execution immediately after the interrupt is handled, this is:

- ✔ ○ a mode switch
- ○ a process switch
- ○ Choose this if none of the others are correct
- ○ a control switch
- ○ an interrupt switch

Question 12 **0.75 / 1 point**

Which is true in a pure KLT facility? (select all that apply)

- ✓ ☐ multithreading is achieved using an application-level threads library
- ✓ ☐ all thread management is done by the application
- ➡ ✓ ☐ the kernel is aware of the existence of threads
- ➡ ✗ ☐ each user-level thread is mapped to a unique kernel-level thread

Question 13 **1 / 1 point**

An event that commonly leads to the termination of a process is: (select all that apply)

- ✓ ☐ A user logs out of an interactive system.
- ✓ ☐ It has exhausted its current time slice in an interactive system.
- ✓ ☐ All of its child processes have terminated.
- ✓ ☐ A user quits an application in an interactive system.

Question 14 **1 / 1 point**

Pure User Level Threads are executing in a multiprogramming, uniprocessing environment. Process P comprises 2 threads, T1 and T2. Initial states are given in the table below. An entity does some action, as specified in the table. What are the resulting states?

P	T1	T2	Action
Running	Ready	Running	T2 requests an action from T1

✔ __1__ P
✔ __1__ T1
✔ __3__ T2

1. Running
2. Ready
3. Blocked
4. Suspended

Question 15 **1 / 1 point**

Some number of user programs are simultaneously submitted for execution in a uniprocessing, multiprogramming system with no virtual memory, no paging, and round-robin scheduling.

The 37 instruction cycles below show interleaved traces from the processor's point of view, starting at the start of the first user program to execute. No user program has terminated by cycle 37.

How many times did the dispatcher execute?

1.	2000	**11.**	501	**21.**	6037	**31.**	500
2.	2001	**12.**	502	**22.**	6038	**32**	501
3.	500	**13.**	3050	**23.**	6039	**33.**	502
4.	501	**14.**	3051	**24.**	500	**34.**	6040
5.	502	**15.**	3052	**25.**	501	**35.**	6041
6.	6032	**16.**	3053	**26.**	502	**36.**	6042
7.	6033	**17.**	500	**27.**	3054	**37.**	6043
8.	6034	**18.**	501	**28.**	3055		
9.	6035	**19.**	502	**29.**	3056		
10.	500	**20**	6036	**30.**	3057		

○ 1
○ 2
○ 3
○ 4
✔ ○ 5
○ 6

- 7
- 8
- 9

Question 16 **1 / 1 point**

The traditional approach of exactly one thread of execution per process, in which the concept of a thread is not recognized, is referred to as a _____ .

- multiprocessing approach
- non-threaded approach
- Choose this if none of the others are correct
- lightweight process approach
- ✔ single-threaded approach

Question 17 **3.2 / 4 points**

The table below shows relevant events (and the time each occurs) in a multiprogramming, single processor system. At time 0 all processes are ready or running and the only resources in use are processor and memory.

Use the dropdowns to choose the state of each process at time 36.

Time	Event
5	P3's timeslice ends
8	P1 requests to read from disk
11	P2 waits for a signal
14	P3 spawns P4

17	P4 requests to read from disk
20	P1 is swapped out
23	P5 is swapped out
26	P3's timeslice ends
29	Interrupt: P2 receives signal
32	P1 swapped back in
35	P2's time slice ends

✓	__1__	P2	**1**. ready/running
✓	__1__	P3	**2**. blocked
✓	__2__	P4	**3**. blocked/suspend
✗	__1__ (2)	P1	**4**. ready/suspend
✓	__4__	P5	**5**. exit

Question 18 **1 / 1 point**

Why might a process transition from state Running to Blocked according to our text? (select all that apply).

- ✓ ☑ the process has reached the maximum allowable time for uninterrupted execution
- ✓ ☑ the process waits for another process to provide data
- ✓ ☑ the process requests a resource that is not immediately available
- ✓ ☑ to free up a sufficiently large block of main memory

Question 19 **0 / 1 point**

The purpose of jacketing is:

- ✗ ◯ to convert a ULT into a KLT
- ➡ ◯ to convert a blocking system call into a nonblocking system call
- ◯ to convert a nonblocking system call into a blocking system call
- ◯ to convert a KLT into a ULT
- ◯ Choose this if none of the others is correct

Question 20 **1 / 1 point**

Pure User Level Threads are executing in a multiprogramming, uniprocessing environment. Process P comprises 2 threads, T1 and T2. Initial states are given in the table below. An entity does some action, as specified in the table. What are the resulting states?

P	T1	T2	Action
Blocked	Ready	Running	Interrupt: T2's disk read completes

- ✓ __2__ P
- ✓ __2__ T1
- ✓ __1__ T2

1. Running
2. Ready
3. Blocked
4. Suspended

Question 21 **1 / 1 point**

In a multithreaded environment, a _____ is defined as the unit of resource allocation and a unit of protection.

- ◯ Choose this if none of the others are correct
- ◯ trace
- ✓ ◯ process
- ◯ program
- ◯ thread

Done