Q

Terminology

What is Preemption?

Q

Preemption

What new challenges did Preemption introduce when compared to cooperative multiprogramming?

Q

Preemption

How is Preemption implemented in an operating system kernel?

Q

Terminology

Compare Concurrency and Parallelism.

Q

Terminology

What is a Critical Section?

Q

Concurrency

What is a the value of shared, and why?

```
int shared = 0;
void func_a(void)
{
   shared++;
}
```

```
void func_b(void)
{
   shared--;
}
```

Q

Concurrency

Describe the three criteria that correct solutions of the Critical Section problem must fulfill.

Q

Concurrency

Why does this naive approach not solve the critical section problem? Outline a schedule.

```
shared int turn = 0;
do { // Code for T_i
    while (turn != i);
        critical section
    turn = j;
        remainder section
} while (1);
```

Q

Concurrency

Name and describe one well known software algorithm that solves the critical section problem.

Q

Concurrency

Discuss whether pure software algorithms are a good solution to the critical section problem.

Q

Concurrency

To mitigate the problems of the software algorithms, hardware approaches are used instead. Which is not one of them?

a) test-and-set

b) outlook

c) exchange

d) interrupt disabling

Q

Synchronization

What is a Semaphore? What operations are defined on Semaphores?

Q

Synchronization

Identify the advantages and disadvantages of using Semaphores over native hardware approaches.

Q

Synchronization

What is a Deadlock? Describe how a Deadlock can be the result of careless use of Semaphores.

Q

Synchronization

Given the following producer/consumer example program, how could you guard it with semaphores?

```
void producer (void) {
  int item;
  while (1) {
      produce_item(&item);
      if (count == N) suspend();
      insert_item(item);
      count = count+1;
      if (count == 1) wake(consumer)
    }
}
```

```
void consumer (void) {
  int item;
  while (1) {
      if (count == 0) suspend();
      remove_item(&item);
      count = count-1;
      if (count == N-1) wake(producer);
      consume_item(item);
    }
}
```

Q

Windows Synchronization

Describe the Role of the Dispatcher Objects.

Q

Windows Synchronization

What does it mean for a Dispatcher Object to be Signaled, or Non-Signaled?

Describe for one type of dispatcher object what can cause it to change between these states.

Q

Synchronization

Name two functions of the Windows or UNIX API related to process synchronization and explain their purpose.

Q

Interrupts

What is the IRQL?

Q

Interrupts

Describe what tasks the operating system handles at various ranges of IRQLs.

Q

Interrupts

What is a Trap?

Q

Interrupts

What kinds of event can cause a Trap?

Q

Interrupts

What happens if during interrupt processing an interrupt of lower precedence arrives? What if the precedence is higher?

Q

Interrupts

What are the roles of the Deferred and Asynchronous Procedure Calls? (DPC and APC)

Q

Interrupts

A Thread on Windows executes the Function ReadFile(). Describe the flow of activity from the application to the device and back.

(you can get bonus points for a great answer)

Q

Inter-Process Communication

What is a pipe? Why is it needed?

Q

Inter-Process Communication

What is the difference between a pipe and a socket?

Q

Inter-Process Communication

Pipes and sockets are tied to the lifetime of a process. What other UNIX construct could you use that survives its creating process, and where is it persisted?

Q

Inter-Process Communication

Write in Pseudocode (Windows or UNIX semantics) a program that launches two processes and connects them through a pipe.

Q

Inter-Process Communication

Name two functions of the Windows or UNIX API related to inter-process communication, and explain their purpose.

Q

What?

bs@hpi.de