

# STUDY QUESTIONS FOR COURSE CS570 (OPERATING SYSTEMS)

(Chapter 1 is missing intentionally.)

## Chapter 2: Program

1. Explain the following terms:
  - (a) Object module
  - (b) Load module
  - (c) Image of the load module
2. Dynamic linking
  - (a) What is the difference between dynamic and static linking
  - (b) What kind of dynamic linking do you know
  - (c) Claim at least three advantages of dynamic linking
3. Implementation of dynamic linking. Suppose the following modules are given:

```
Module X:                Module Y:
  Export A, B;            Export C;
  Import C;              Import: None;
  Function A()           function D(){...}
  {                      function C()
    {                    {
      call A();          {
      call B();          call D();
      call C();          }
    }                    }
  }
  function B(){...}
```

Show the general structure of the corresponding object modules. Indicate data structures and draw the linkages (pointers) before (solid lines) and after dynamic linking (dashed lines).

4. Write a segment of C++ code in which you load a dynamic link library ddd.dll which has a function defined as:

```
double fff (char *x){...}
```

After you load the library, call the function via pointer named ppp. What you need to do before you call the function via pointer?

5. What are the differences between load-time and run-time dynamic linking?
6. Explain sharing of dynamic link libraries between two programs. (Use diagrams)
7. What is software trap and how is it used in operating system design?

8. Explain major difference between system and user mode.
9. How is the system mode (supervisory mode) characterized in hardware?
10. Describe what happens when CPU executes TRAP and RET instructions (RET means return from exception, i.e. return from trap)

### **Chapter 3: Processes and Threads**

1. What is the fundamental difference between multiprogrammed batch system and time sharing system?
2. What constitutes a process?
3. What is process control block? What information does it contain?
4. Draw the process/thread state diagram (use five-state model).
5. What causes a process/thread to change the state:
  - (a) From running to ready?
  - (b) From ready to running?
  - (c) From running to blocked?
  - (d) From blocked to ready?
6. What happens when some thread changes its priority (suppose a prioritized thread scheduling)?
7. How many thread priorities has Windows NT?
8. What parts of a process its threads can share?
9. Does threads have their own heap and/or stack? Justify your answer.
10. Why threads must have their own stack?
11. Why processes don't have stack (in a threaded operating system)?
12. What is time quantum in a time sharing system?
13. What is the difference between process context and thread context switch in a threaded system?
14. What is difference between preemptive and non-preemptive scheduling?
15. List at least three examples that justify usage of threads.
16. Is there any difference between thread switching if threads belong

or do not belong to the same process?

17. On the same diagram sketch the histogram of CPU burst durations for CPU-bound and I/O-bound threads.
18. Describe terms long-term, medium-term and short-term thread scheduling.
19. Explain terms CPU utilization, throughput, turnaround time, waiting time and response time. (Which scheduling criteria is used in multiprogrammed batch systems and which in time sharing systems?)
20. List all process/thread scheduling policies you know.
21. Suppose five threads are waiting in a ready queue. Their next CPU burst times are as follows:  
5 (at head), 8, 10, 2, 4 (at tail)  
Compute the average waiting time for the following scheduling policies:  
(a) FCFS (first come first served)  
(b) SJF (shortest job first)  
(c) RR (round robin, time quantum is 3 time units)
22. Explain how the exponential averaging works in the approximation of the SJF scheduling algorithm.
23. Comment the fairness problem in RR scheduling. What approaches you know for improvement of fairness of RR scheduling policy?
24. Compare symmetric and asymmetric multiprocessing. What kind of multiprocessing is used in Windows NT?

## **Chapter 4: Memory Management**

1. Describe the terms: internal and external fragmentation.
2. Compare the following algorithms in terms of external and internal fragmentation: fixed-size partitioning, dynamic partitioning, paging, segmentation, segmentation with paging.
3. What are the major tasks of memory management?
4. What memory management schemes do you know?
5. Describe fixed-size partitioning. Give:  
(a) Schematic diagram  
(b) Address translation equation  
(c) How can be problem of internal fragmentation lessened?
6. Describe dynamic partitioning. What is the main problem in dynamic partitioning? How is it solved?

7. Describe the concept of paging. Use a simple diagram with two processes and memory.
8. What are consequences of choosing large page sizes, and what of small page sizes?
9. Why the page size has to be a power of two?
10. What is the conceptual difference between fixed-size partitioning and paging?
11. Explain address translation of one level paging scheme:
  - (a) Draw the block diagram, show all important registers and data structures
  - (b) Give address translation equation
  - (c) How many pages has a system with 16-bit addressing, with page size 512 bits (give detailed calculations, not just the final result.)
12. What is translation look-aside buffer?
  - (a) Why is it used
  - (b) How many CPU cycles does it take to search the TLB and why?
  - (c) How many TLB register has Intel 80486?
  - (d) What is TLB hit-ratio?
13. What would be the average address translation time of one-level paging system with 85% hit ratio. Suppose that the access times of TLB and internal memory are 15 ns and 80 ns respectively.
14. What protection bits in a paging system you know? Where are they residing?
15. Describe a page table entry, what does it contain and what is the purpose of its fields?
16. Describe two-level paging scheme
  - (a) Draw the block diagram (use one block for the part that does one-level paging)
  - (b) Give address translation equation
17. Suppose a 32-bit addressing system with 4 KB pages. Answer the following:
  - (a) Show the structure of the logical address
  - (b) How many page directories has the system?
  - (c) What is the size of page directory entries?
  - (d) How many page tables has the system?
  - (e) What is the size of page tables?
  - (f) How many pages is required for each page directory or page table?
  - (g) Describe page table and page directory entries (size, fields)
18. Suppose a logical address  $LA = (8400901)_{10}$  and 4 KB pages. Derive the directory number and the page number.
19. Describe address translation with inverted page tables.

- (a) Give a block diagram that shows the major components.
- (b) What are advantages of inverted page tables?
- (c) What are disadvantages of inverted page tables?
- (d) How is the efficiency of inverted page tables improved?

20. Describe segmentation

- (a) Give a block diagram that shows the major components.
- (b) Describe entries of segment tables (fields and their functions)
- (c) What rationale is behind segmentation.
- (d) What is advantage of segmentation?
- (e) What is disadvantage of segmentation?
- (f) How can the major disadvantage of segmentation be removed?

## Chapter 5: Virtual Memory

1. What is demand paging? What is alternative to demand paging?
2. What is the major task of virtual memory?
3. What is page fault?
  - (a) When a page fault occurs?
  - (b) How many times can a page fault occur in the case of a tree-address instruction?
  - (c) Describe the actions of a page fault handler (give a simplified pseudo-code)
  - (d) What is page fault rate?
4. What are the potential problems with restarting a machine instruction after a page fault?
5. What is page replacement? List all page replacement policies you know.
6. Shortly describe FIFO algorithm for page replacement. What are its advantages and disadvantages?
7. Shortly describe LRU algorithm for page replacement. What are its advantages and disadvantages?
8. What is Belady's anomaly?
9. Shortly describe Clock algorithm for page replacement.
10. Compare global and local replacement algorithms.
11. What is principle of locality (in the context of virtual memory)?
12. What is thrashing?
13. Does the process utilization always increase if you increase the

degree of multiprogramming? Justify your answer.

14. What is working set:
  - (a) Give informal definition
  - (b) Give formal definition
  - (c) Who has introduced the idea?
  - (d) On what basic principle is the idea based?
15. What is working set model in virtual memory management?
16. Explain the implementation of the working set model based on page fault rate.
17. What is working set trimming? In what operating system is it used?
18. What is page locking? Give at least one example of its usage.

## Chapter 6: Thread Synchronization

1. What is critical section? Give an example different than in Lecture Notes.
2. What is mutual exclusion?
3. Compare busy-waiting and thread blocking approaches in implementation of mutual exclusion.
4. Can direct disabling/enabling of interrupts be used to implement mutual exclusion? Justify your answer.
5. Show why and how is busy waiting supported in hardware.
6. What the hardware support for busy waiting has to provide in case of multiprocessors?
7. Give the pseudocode for ENTER and LEAVE trap handlers for multiprocessor systems (use block(x) and unblock(x) macros). Suppose the following structure associated with the synchronization object: `typedef struct {BOOL m; QUEUE b; BOOL s; /*for spinlock*/} CS;` Comment the code!
8. Give an example of mutual exclusion with enter() and leave() APIs (show consumer and producer threads which enter a critical section.)
9. Write a segment of code that enters and exits a critical section by using Win32 APIs for critical section.
10. What is deadlock? Give an example of incorrect usage of mutual exclusion which causes a deadlock.
11. Who has invented semaphores?
12. Explain the semantics of P() and V() semaphore operations.
13. Give the pseudocode for P and V trap handlers for multiprocessor systems (use block(x) and unblock(x) macros). Suppose the following structure associated with the synchronization object: `typedef struct {int a; QUEUE b; BOOL s; /*for spinlock*/} CS;` Comment the code!

14. Give an example of mutual exclusion with semaphores (show consumer and producer threads which enter a critical section and use P() and V() operations)
15. Suppose there are three output devices which can be used by a multithreaded program by calling a function OUTPUT(). Write a segment of code which is calling OUTPUT() only if there are free devices. How many semaphores are required for this? Use Win32 semaphores. Show the code for the main program and for the threads (do not forget to initialize semaphore(s)).
16. What are events?
17. Give the pseudocode for WAITEVENT and SETEVENT trap handlers for multiprocessor systems (use block(x) for WAITEVENT, and runing2ready(), blocked2ready(x) and ready2running() for SETEVENT.) Suppose the following structure associated with the synchronization object:
 

```
typedef struct {BOOL m; QUEUE b; BOOL s; /*for spinlock*/} CS;
```

 Comment the code!

## Chapter 7: Input-Output

1. Describe the general I/O architecture (give a block diagram and name all important components)
2. What is device controller?
3. What is device driver?
4. What is I/O manager?
5. What is application programming interface (in the context of I/O)?
6. Describe shortly the basic approaches in I/O:
  - (a) Programmed I/O
  - (b) Interrupt-driven I/O
  - (c) Direct Memory Access
7. Give two block diagrams of an I/O system in which DMA steals one, and two bus cycles per word transfer.
8. List all character-oriented devices you know.
9. List all block-oriented devices you know.
10. What is I/O buffering (why is it used and how does it work)?
11. Describe disk storage geometry.
12. What is disk formatting and what basic information is stored with each disk sector.
13. What is disk blocking and why is it used?

14. Explain terms: virtual block number (VBN) and logical block number (LBN). Who is mapping VBN into LBN?
15. What is disk scheduling?
  - (a) When is it used?
  - (b) Why is it used?
  - (c) Who is performing the disk scheduling?
  - (d) What disk scheduling algorithms do you know?
16. Describe shortly FCFS disk scheduling algorithm and list its advantages and disadvantages.
17. Describe shortly SSFT disk scheduling algorithm and list its advantages and disadvantages.
18. Describe shortly SCAN disk scheduling algorithm and list its advantages and disadvantages.
19. Describe shortly C-SCAN disk scheduling algorithm and claim its advantage over SCAN.

## **Chapter 8: File System**

1. Describe terms sequential and direct file access.
2. What is file allocation? List all the file allocation approaches you know.
3. Describe contiguous file allocation. Use block diagrams. Advantages and disadvantages?
4. Describe linked file allocation. Use block diagrams. Advantages and disadvantages?
5. Describe indexed file allocation. (include single level and multilevel indexed allocation). Use block diagrams. Advantages and disadvantages?
6. Describe combined multilevel indexed allocation. Use block diagrams. Advantages and disadvantages?
7. Describe combined indexed and contiguous allocation. Use block diagrams. Advantages and disadvantages?
8. What is free space management in a file system? What approaches do you know?
9. Describe linked list approach in free space management. Use block diagrams. Advantages and disadvantages?
10. Describe indexing and linking in free space management. Use block diagrams. Advantages and disadvantages?

11. Describe bit map approach in free space management. Use block diagrams. Advantages and disadvantages?
12. What is directory in a file system? What is its function?
13. What is file header and what information does it contain (try to remember as much as you can)?
14. How is file header called in UNIX family of operating systems?
15. Why is desirable to separate directories and file headers?
16. What are hard links in a file system?
17. What are soft links in a file system?
18. How many disk blocks can address 16-bit FAT? What is the maximum disk capacity if disk blocks have size of 512 bytes?
19. What is the maximum capacity of the disk space in case of 32-bit FAT with 2 Kbyte blocks.
20. Describe terms file descriptor, file table and file header. What is their relationship and where are they residing (user space, system space)?
21. Why are file tables separate from file header?

## **Chapter 9: Interprocess Communication**

1. Why threads that belong to different processes cannot communicate through global variables like thread that belong to the same process?
2. What IPC methods do you know?
3. List the function calls used in UNIX shared memory.
4. How is shared memory used in Windows 95/NT?
5. Write a segment of code for two threads, T1 and T2 , which belong to different processes and which access members x and y of the same structure s defined as: `struct {int x; double y;} s`. Use Win32 APIs.
6. What are messages? Describe semantics of message primitives `send()` and `receive()` in terms of thread blocking.
7. Which IPC method is more efficient shared memory or message passing? Justify your answer.
8. What are UNIX System V messages? List the system calls.

9. What are pipes? How are they generally implemented? (Describe both, unnamed and named pipes)
10. What is restriction of Win32 unnamed pipes? Compare with named pipes.
11. Can unrelated processes communicate with unnamed pipes? Justify your answer.
12. List all properties of named pipes in Win32.
13. Write a code for a server which creates a named pipe, waits for a connection request from a client and creates client thread. Also, give the code for the client thread which waits for a message from the client, then answers it. (Use Win32 pipes.)
14. Write a code for a client which connects to a server via named pipe, then sends a message to the server and waits for an answer.
15. What is the main advantage of sockets over Win32 pipes?
16. List the function calls for sockets used by servers. Describe shortly what is the function of each of the calls.
17. List the function calls for sockets used by clients. Describe shortly what is the function of each of the calls.

*(The list of questions will be finished later)*