(Total time 90 minutes, Total Points = 100 points)

Name: (please print) _____

This examination allows **only one page A4 cheat sheet, absolutely no notes and close book**. You may not collaborate in any manner in the exam. You are not allowed to use any means to copy the exam book/question(s). In recognition of and in the sprit of the Oakland University Honor Code, I certify that I will neither give nor receive aid during the exam.

Signature: _____

Hints:

1. Put your name on the exam books NOW!
2. Read the questions clearly and think it through before you are answering.
3. You have 90 minutes to complete the exam. Be a smart exam taker! Remember not all the points born equal. So, if you get stuck on one problem go on to another problem.
4. If you think your answer is not provided as a choice (it will rarely happen), put your answer along with the question.
5. Put your final answers in the answer sheet on the second page.

Answer sheet

Question 1 (Total 37 points)					
1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17.1	17.2
					Earned:_____

Question 2 (Total 27 points)					
1	2	3	4	5	6
7	8	9			
					Earned:_____

Question 3 (Total 24 points)					
1	2	3	4	5	6
7	8				
					Earned:_____

Question 4 (Total 12 points)				
1	2	3	4	
				Earned:_____

Total: _____

Question 1. (37 points) [2 points are deducted for each wrong answer]

1. From a waiting state, a process **<u>can only</u>** enter into _____

A. running state
B. ready state
C. new state
D. terminated state

Answer: _____

2. The full form of PCB is: _____

A. Public Control Block
B. Process Control Box
C. Process Creating Block
D. Process Control Block

Answer: _____

3. Each process has its own PCB. A. True B. False

Answer: _____

4. There can be more than one process in running state at any given time for a single CPU computer system A. True B. False

Answer: _____

5. The state of a process is stored in its _____.

A. registers
B. PCB
C. source code
D. memory

Answer: _____

6. All the processes which are ready to execute reside in _____.

A. I/O queue
B. waiting queue
C. ready queue
D. running queue

Answer: _____

7. What is the function of short-term scheduler?

A. selects a process from secondary storage device and allocates it to the CPU.
B. selects a process from memory and swaps out to secondary storage.
C. selects a process from ready queue and assigns it to the CPU.
D. selects a process from I/O queue to moves it to ready queue.

Answer: _____

8. The function of long-term scheduler is to:
A. move the process from secondary storage to ready queue.
B. move the process from ready queue to CPU.
C. move the process from memory to secondary storage.
D. move the process between different queues.

Answer: _____

9. What is the function of mid-term scheduler?
A. It moves the process from ready queue to CPU.
B. It swaps out the idle process from memory to secondary storage.
C. It moves the process between different queues.
D. It helps the CPU in executing the process.

Answer: _____

10. Which scheduler maintains the Degree of Multiprogramming?

A. Short-Term Scheduler
B. Mid-Term Scheduler
C. Long-Term Scheduler
D. None of the Above

Answer: _____

11. The switching of CPU between different processes is called _____.

A. Swapping
B. Organizing
C. Context Switching
D. Multiple Switching

Answer: _____

12. Which of the following scheduling algorithms use Time Quantum?
A. FCFS
B. SJF
C. Round Robin
D. Priority Scheduling

Answer: _____

13. One of the major problem with priority scheduling is:
A. Swapping
B. Context Switching
C. Aging
D. Starvation

Answer: _____

14. The processes are executed in the following manner in Round Robin Algorithm.

A. The process coming first is executed first without preemption.
B. The processes are executed according to their priority.
C. The process having the smallest burst time is executed first.
D. The process is executed for a time quantum and when the time quantum expires, the process enters into ready state.

Answer: _____

15. A higher time-quantum in the Round Robin increases the overhead and the response time.

A. True B. False

16. As the ratio of time quantum to job length decreases, round-robin scheduling becomes equivalent to first-come-first-served.

A. True B. False

Answer: _____

17. As shown with the following code, four processes produce output using the routine "printf" and synchronize using three semaphores "R", "S" and "T."

```
Semaphore R=1, S = 3, T = 0; /* initialization */

/* process 1 */    /* process 2 */       /* process 3 */       /*process 4 */
while(true) {      while(true) {          while(true) {         while(true) {
    P(S);              P(T);                  P(T);                     P(R);
    printf('A');       printf ('B');          printf ('D');             printf ('E');
}                      printf ('C');          V(R);                     V(T);
                       V(T);              }                     }
                   }
```

17.1 How many A and B's are printed when this set of processes runs?
A. 2 As and 3 B
B. 3 As and any Bs
C. 1 A and 0 B
D. Any As and any Bs

Answer: _____

17.2 Which is the following output sequence is possible when this set of processes runs?
A. AEBCBCDAA
B. AAABCBCBE
C. AAEBCDEAA
D. EDEDBCAAA

Answer: _____

Question 2. (27 points) [**Process Management**] Considering the program name `fork.c` and answer the following questions [each wrong answer will bring down 3 points]

```
#include  <stdio.h>
#include  <string.h>
#include  <sys/types.h>


int a = 0;

void  main(void)
{
     int    i=1;

     a++;
     sprintf(buf, "A %d\n", a);
     write(1, buf, strlen(buf));

     fork();

     a++;
     sprintf(buf, "B %d\n", a);
     write(1, buf, strlen(buf));

     fork();

     a++;
     sprintf(buf, "C %d\n", a);
     write(1, buf, strlen(buf));

}
```

A user compiles the file from command line >gcc fork.c, an executable file a.out is created, then the user runs the executable from command line. Answer the following questions based on the understanding of the code and execution.

1. How many letter 'A' in total are printed in the output trace?

   A. 1 B. 2 C. 4 D. 7

   Answer : _____ or _____ if you come up with a different answer

2. How many letter 'B' in total are printed in the output trace?

   A. 1 B. 2 C. 4 D. 7

   Answer : _____ or _____ if you come up with a different answer

3. How many letter ‘C’ in total are printed in the output trace?

   A. 1 B. 2 C. 4 D. 7

   Answer : _____ or _____ if you come up with a different answer

4. Where the variable **a** is stored in the process’ address space?

   A. Text segment B. Data segment C. Heap segment D. Stack segment

   Answer : _____ or _____ if you come up with a different answer

5. Where the variable **i** is stored in the process’ address space?

   A. Text segment B. Data segment C. Heap segment D. Stack segment

   Answer : _____ or _____ if you come up with a different answer

6. In the print trace (output string) what number is following ‘A’ ?

   A. 0 B. 1 C. 2 D. 3

   Answer : _____ or _____ if you come up with a different answer

7. In the print trace (output string) what number is following ‘B’ ?

   A. 0 B. 1 C. 2 D. 3

   Answer : _____ or _____ if you come up with a different answer

8. In the print trace (output string) what number is following ‘C’ ?

   A. 0 B. 1 C. 2 D. 3

   Answer : _____ or _____ if you come up with a different answer

Question 3. (24 points) [**CPU Scheduling,** 5 points for each correct answer]
In a computer system, there are 5 processes P_0 through P_4. Their arrival time and CPU burst time are shown as follows. [**note**: when there is a tie in any scheduling, if one of the processes is the current running process then continue to schedule it to avoid context switch, otherwise choose the one with smaller subscript]

Process	Arrival Time	Burst Time
P_0	2.0	3
P_1	0.0	7
P_2	2.0	4
P_3	4.0	1
P_4	5.0	4

1. If the scheduling algorithm is FCFS, what is the order of schedule?

   A. P_0 , P_1 , P_2 , P_3 , P_4
   B. P_1 , P_0 , P_2 , P_3 ,P_4
   C. P_2 , P_1 , P_0 , P_3 ,P_4
   D. P_3 , P_1 , P_2 , P_0 ,P_4

   Answer:_____ or _____ if you come up with a different answer

2. If the scheduling algorithm is FCFS, what is the total waiting time?

   A. 31 B. 32 C. 33 D. 34

   Answer: _____ or _____ if you come up with a different number

3. If the scheduling algorithm is Round Robin with a time quantum of 3, what is the total waiting time?

   A. 41 B. 42 C. 43 D. 44

   Answer: _____ or _____ if you come up with a different number

4. Which process will be scheduled to run first when the scheduling algorithm is preemptive shortest job first?

   A. P_0 B. P_1 C. P_2 D. P_3 E. P_4

   Answer:_____ or _____ if you come up with a different answer

5. Which process will be the second to run when the scheduling algorithm is **non-preemptive** shortest job first?

   A. P_0 B. P_1 C. P_2 D. P_3 E. P_4

   Answer:_____ or _____ if you come up with a different answer

6. Which process will be the second to run when the scheduling algorithm is **preemptive** shortest job first?

   A. P_0 B. P_1 C. P_2 D. P_3 E. P_4

   Answer:_____ or _____ if you come up with a different answer

7. If the scheduling algorithm is non-preemptive shortest job first, what is the total waiting time?

   A. 22 B. 24 C. 26 D. 28

   Answer:_____ or _____ if you come up with a different number

8. If the scheduling algorithm is preemptive shortest job first, what is the total waiting time?

   A. 22 B. 24 C. 26 D. 28

   Answer:_____ or _____ if you come up with a different number

Question 4. (12 points) [**Synchronization**] Suppose that a shared virtual classroom supports students in two majors, **CS** and **ME**. The semantics of this virtual classroom operation allow multiple students of the same major to use the classroom concurrently, but not the different major using the classroom concurrently. Consider the following pseudo codes for the implementations of mutual exclusion protocols for the two types of students that implement the semantics described above.

```
semaphore mutex, lock;
int counter_cs = 0, counter_me = 0;
```

```
Class CS:
A. P(mutex);
B. counter_cs:= counter_cs + 1;
C. if (counter_cs == 1)
D.     P(lock);
E. V(mutex);
F. … …
G. Critical Section
H. … …
I. P(mutex);
J. counter_cs := counter_cs - 1;
K. if (counter_cs == 0)
L.     V(lock);
M. V(mutex);
```

```
Class ME:
A. P(mutex);
B. counter_me := counter_me + 1;
C. if (counter_me == 1)
D.     P(lock);
E. V(mutex);
F. … …
G. Critical Section
H. … …
I. P(mutex);
J. counter_me := counter_me - 1;
K. if (counter_me == 0)
L.     V(lock);
M. V(mutex);
```

1. In the above implementation, if we initialize mutex = 0, lock = 0, which of the following statements is correct?

   A. There is no progress.
   B. There is deadlock.
   C. There is violation on mutual exclusive access.
   D. It is a correct implementation.

   Answer: _____ or _____

2. In the above implementation, if we initialize mutex = 1, lock = 1, which of the following statements is correct?

   A. There could be 5 CS students and 1 ME student in the classroom the same time.
   B. There could be 5 CS students in the classroom while 10 ME students waiting outside of the classroom.
   C. When there are 5 CS students in the classroom while 5 ME students want to enter, there would be a deadlock occurring.
   D. Both CS and ME students can't enter the classroom for this implementation.

   Answer: _____ or _____

3. In the above implementation, if we initialize mutex = 1, lock = 1, while there are 5 CS students in the classroom, there are 5 ME students come and want to enter the classroom, where the ME students would be blocked at?

   A. They are all blocked at `line A`.
   B. They are all blocked at `line F`.
   C. The first is blocked at `line D`, the other four are blocked at `line A`.
   D. The first is blocked at `line D`, the other four are blocked at `line F`.

   Answer: _____ or _____

4. A new version of virtual classroom is implemented as follows, which of the following statements is correct?

```
semaphore mutex_cs=1, mutex_me = 1, lock=1;
int counter_cs = 0, counter_me = 0;
```

```
Class CS:                              Class ME:
N. P(mutex_cs);                        N. P(mutex_me);
O. counter_cs:= counter_cs + 1;        O. counter_me := counter_me + 1;
P. if (counter_cs == 1)                P. if (counter_me == 1)
Q.    P(lock);                         Q.     P(lock);
R. V(mutex_cs);                        R. V(mutex_me);
S. … …                                 S. … …
T. Critical Section                    T. Critical Section
U. … …                                 U. … …
V. P(mutex_cs);                        V. P(mutex_me);
W. counter_cs := counter_cs - 1;       W. counter_me := counter_me - 1;
X. if (counter_cs == 0)                X. if (counter_me == 0)
Y.    V(lock);                         Y.     V(lock);
Z. V(mutex_cs);                        Z. V(mutex_me);
```

   A. The new version avoids the starvation.
   B. The new version satisfies the mutual exclusion.
   C. The new version may cause deadlock.
   D. The new version may cause more delay in counting.

   Answer: _____ or _____

(THE END)