***“I pledge that the answers in this exam/quiz are my own and that I will***

***not seek or obtain an unfair advantage in producing these answers.***

***Specifically,***

* ***I will not plagiarize (copy without citation) from any source;***
* ***I will not communicate or attempt to communicate with any other***

***person during the exam/quiz; neither will I give or attempt to give***

***assistance to another student taking the exam/quiz; and***

* ***I will use only approved devices (e.g., calculators) and/or approved***

***device models.***

***I understand that any act of academic dishonesty can lead to disciplinary***

***action.”***

Name:

SID:

Section A:

1. I) D

II) C

1. A
2. I) High

II) Middle

III) Low

IV) Low

V) Middle

VI) High

1. V -> III -> I -> IV -> II
2. C
3. I) Running to Ready

II) Ready to Ready/Suspend

III) Running to Blocked

1. Process control block information
2. A
3. B
4. D
5. C
6. I) Both

II) Process

1. Mutual exclusion
2. A
3. B
4. D
5. A
6. B
7. 3
8. A
9. A
10. A
11. C
12. D

Section B:

1. I)

Process2: 2 4

Thread: 3 4

Process1: 2 5

II)

prog\_global = 3, when the program starts.

prog\_global = 4, in Process2 as child read the value as 3 and increased it by 1.

prog\_global = 4, in Thread as parent read the value as 3 and thread increased it by 1.

prog\_global = 5,

in Process1 as parent receive value as 4 from thread and increased it by 1.



|  |  |  |
| --- | --- | --- |
| Process | Line number | x |
| P1 | 1 | 10 |
| P2 | 1 | 10 |
| P1 | 2 | 11 |
| P2 | 2 | 12 |
| P1 | 3 | 11 |
| P1 | 4 | 11 |
| P2 | 3 | 10 |
| P2 | 4 | 10 |
| P1 | 5 | 10 |
|  | “x is 10” | |

1. I)

|  |  |  |
| --- | --- | --- |
| s | n | e |
| 1 | 0 | 10 (all empty) |
| 1 | -1 | 10 |

Synchronization can be achieved by blocking and unblocking, when a consumer wants to take a data item from an empty buffer, n = -1 by semWait(n), consumer will be block and wait for producer to process and unblock by semSignal(n).

II)

|  |  |  |
| --- | --- | --- |
| s | n | e |
| 1 | 0 | 10 (all empty) |
| 1 | -1 | 10 |
| 1 | 0 | 9 |

Synchronization can be achieved by blocking and unblocking, when a producer wants to append a data item to the buffer, when the process is done, n = 0 by semSignal(n), the customer is unblocked.

1. I) sem\_x = 0, sem\_y = 1

II) Blocking by sem\_x as sem\_x = -1

III) Blocking by sem\_y as sem\_y = -1

IV) sem\_x = 0, reader1 will read, when reader1 finish reading, and after reader1 finish reading, reader2 will read.