**Instructions:**

* Only handwritten solution will be accepted.
* Submission must be made in uncompressed format.
* Understanding the problem is part of the problem.

**Question#1:** Answer the following questions in your own words precisely. (4+5+6) Marks

1. Write two advantages and one limitation of using shared memory over anonymous pipes for inter-process communication.
2. How does the creation of multiple processes speed up summing an array containing 1000 elements on a uniprocessor and a multiprocessor machine?
3. Answer the following for a monolithic kernel vs a modular kernel. Fill the table with a Yes or No in each cell.

|  | Monolithic kernel | Modular kernel |
| --- | --- | --- |
| Add new features at run time | NO | YES |
| Add new hardware at run time | NO | YES |
| Add new features (requires recompilation) | YES | NO |
| Remove features at run time | NO | YES |
| Easy to debug errors (where the error occurred) | NO | YES |
| More chances of introducing bugs as compared to other | YES | NO |

| **-** | **P0** | **P1** | **P2** | **Tot** | **Avg** |
| --- | --- | --- | --- | --- | --- |
| **1** | **4** | **7** | **8** | **19** | **6.33333333333333** |
| **2** | **4** | **7** | **8** | **19** | **6.33333333333333** |
| **3** | **0** | **6** | **8** | **14** | **4.66666666666667** |
| **4** | **0** | **7** | **8** | **15** | **5** |
| **5** | **0** | **3** | **8** | **11** | **3.66666666666667** |
| **6** | **0** | **3** | **8** | **11** | **3.66666666666667** |
| **7** | **0** | **3** | **8** | **11** | **3.66666666666667** |
| **8** | **0** | **3** | **8** | **11** | **3.66666666666667** |
| **9** | **0** | **3** | **8** | **11** | **3.66666666666667** |
| **10** | **0** | **3** | **8** | **11** | **3.66666666666667** |

**Question#2:** (10 Marks)

Schedule the given processes using Round-Robin scheduling policy. Also, draw Gantt chart and calculate average wait time. Time slice is the last digit of your registration number + 1.

Example: If your registration# is L1F16BSCS1234, then the time slice will be 4 + 1 => 5.

| **Processes** | **CPU Burst** |
| --- | --- |
| P0 | 3 |
| P1 | 5 |
| P2 | 23 |