Assignment II

Problem Bank 22

Assignment Description:

The assignment aims to provide deeper understanding of Pipelining Architecture, Scheduling and Multithreading using CPU- OS Simulator. The assignment has three parts.

- Part I deals with Pipeline Architecture
- Part II deals with Scheduling algorithm(FCFS, SJF and RR)
- Part III deals with Multithreading

Submission:

You will have to submit this documentation file and the name of the file should be GROUP-NUMBER.pdf. For Example, if your group number is 1, then the file name should be GROUP-1.pdf.

Submit the assignment by **3rd March 2022,** through **CANVAS only**. File submitted by any means outside CANVAS will not be accepted and marked.

In case of any issues, please drop an email to the course TAs, Ms. Michelle Gonsalves (michelle.gonsalves@wilp.bits-pilani.ac.in).

Caution!!!

1. Assignments are designed for individual groups which may look similar and you may not notice minor changes in the assignments. Hence, refrain from copying or sharing documents with others. Any evidence of such practice will attract severe penalty.
2. **Marks will not be awarded for individual submissions**

Evaluation:

- The assignment carries 12 marks
- Grading will depend on
    - Contribution of each student in the implementation of the assignment
    - **Plagiarism or copying will result in -12 marks**

**************************FILL IN THE DETAILS GIVEN BELOW****************

Assignment Set Number:

Problem Bank 22

Group Name: 102

Contribution Table:

Contribution (This table should contain the list of all the students in the group. Clearly mention each student's contribution towards the assignment. Mention "No Contribution" in cases applicable. If the contribution is equal the write 100%)

Sl. No.	Name (as appears in Canvas)	ID NO	Contribution (%)
1	Reena Nagrale	2021fa04035	50%
2	Ankur Gupta	2021fa04040	50%
3	Pratiti Shah	2021fa04031	5%

Resource for Part I, II and III:

- Use following link to login to "eLearn" portal.
  - https://elearn.bits-pilani.ac.in
- Click on "My Virtual Lab – CSIS"
- Using your canvas credentials login in to Virtual lab
- In "BITS Pilani" Virtual lab click on "Resources". Click on "Computer Organization and software systems" course. Refer to LabCapsule 4, LabCapsule 5, LabCapsule 6.

Part I: Pipeline Processor

Consider the following program:

```
program pipeline2
    a=10
    b=20
    c=30
    b=a+b
    c=c+b
    a=a/3
    a=c
end
```

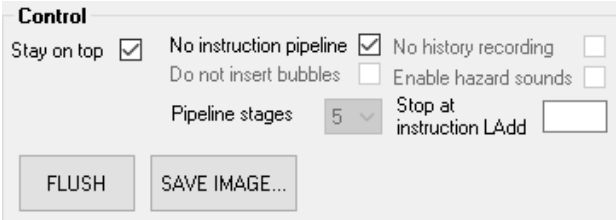
Compile the code and load it in CPU-OS simulator. Perform the following:

Execute the above program using non-pipelined processor and pipelined processor and answer the following questions.

Note: Every time flush the pipeline before running the code

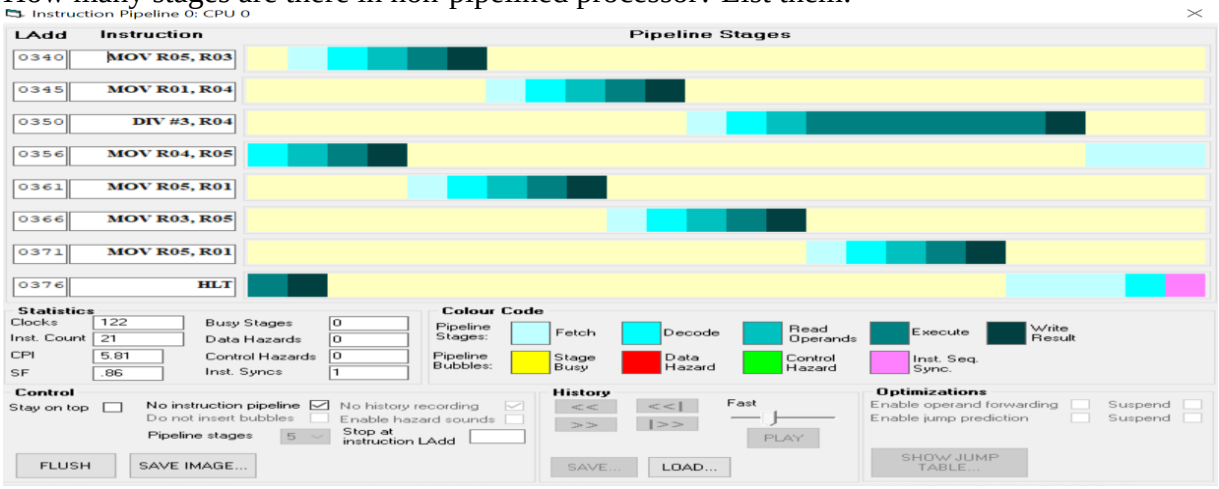
A) Non-pipelined Processor:

To enable non-pipelined processor, check "No instruction pipeline" check box in control panel.



How many stages are there in non-pipelined processor? List them.

a)



Solution: 5 stages – Fetch, Decode, Read Operands, Execute and Write Result

b) Fill in the following after executing of above program using non-pipelined processor.

	Clocks	Instruction Count	CPI	Speed up Factor
Non-Pipelined Processor	**122**	**21**	**5.81**	**.86**

c) What are the contents of General-purpose registers after the execution of the program?

Solution:

R01 – 60
R02 – 30
R03 – 30
R04 – 3
R05 – 60

B) Pipelined processor:

To use, enable pipelined processor, uncheck "No instruction pipeline" check box in control panel.

a) Fill in the following table with respect to pipelined processor execution of the above program:

Fig1 – Pipelined Processor with Do not insert bubble

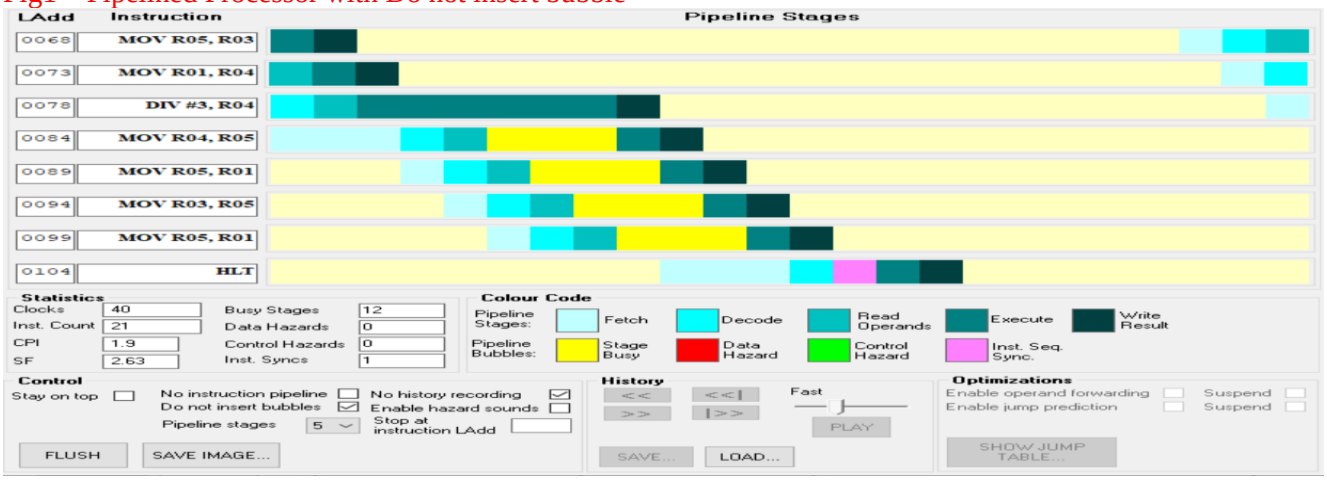
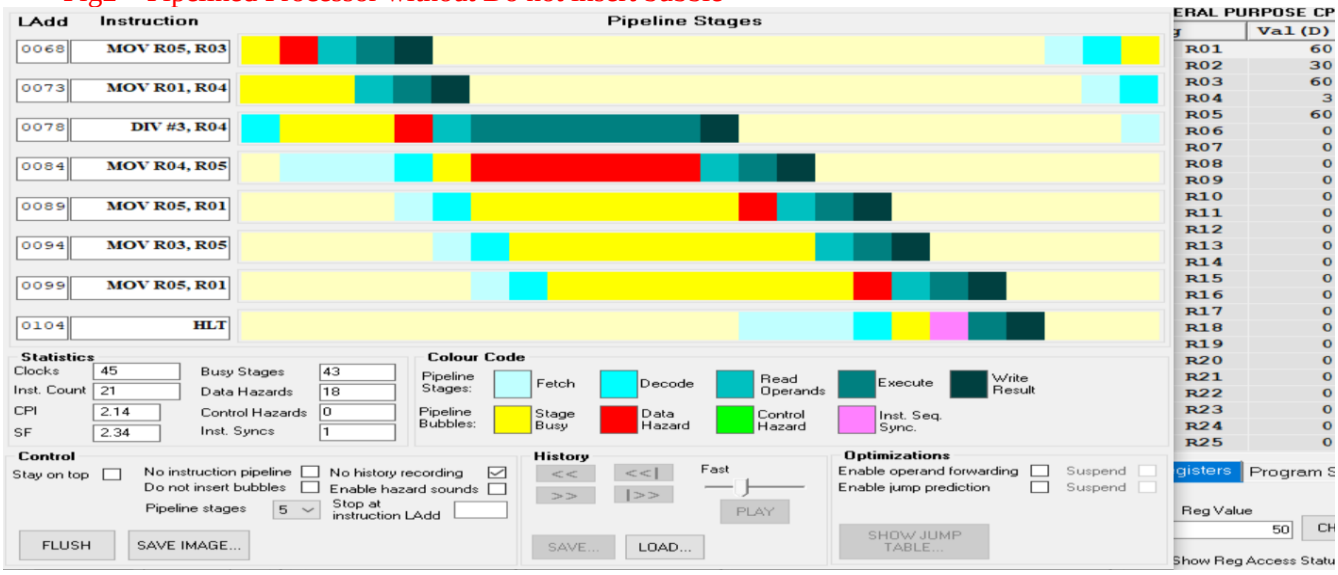


Fig2 – Pipelined Processor without Do not insert bubble



Pipelined processor conditions	**Clocks**	**Instruction Count**	**CPI**	**Speed up Factor**	**Data hazard (Yes/No)**	**Contents of registers used by the program**
Check "Do not insert bubbles" check box	**40**	**21**	**1.9**	**2.63**	**No**	**R01 – 50** **R02 – 30** **R03 – 20** **R04 – 16** **R05 – 20**
Uncheck "Do not insert bubbles"	**45**	**21**	**2.14**	**2.34**	**Yes**	**R01 – 60** **R02 – 30** **R03 – 60** **R04 – 3** **R05 – 60**

b) Is there a way to improve the CPI and Speed up factor? If so give the solution.

Solution:
Enable operand forwarding as optimization and enable jump prediction
The pipeline has a means of forwarding operands, i.e. prior to previous instructions updating registers in a later stage, that contributes to reducing the CPI. There is also the technique of jump predictions that helps to improve CPI and Speedup factor.

Part II: Process Scheduling

Consider the following source codes:

Source Code 1:

```
program My_Pgm1
        i = 1
        for n = 1 to 10
                x = i + n
        next
end
```

Source Code 2:

```
program My_Pgm2
        i = 10
        for n = 1 to 8
                x = i + n
        next
end
```

Source Code 3:

```
program My_Pgm3
        i = 10
        for n = 1 to 15
                x = i + n
        next
end
```

Source Code 4:

```
program My_Pgm4
        i = 10
        for n = 1 to 5
                x = i - n
        next
end
```

Compile the above source code and load it in the main memory.

We are now going to use the OS simulator to run this code. To enter the OS simulator:

1) Click on the OS O… button in the current window. The OS window opens.
2) You should see an entry titled as the program name given above, in the PROGRAM LIST view.
3) Now that this program is available to the OS simulator, we can create as many instances, i.e. processes, of it as we like. You do this by clicking on the CREATE NEW PROCESS button.

PART-II_A

- Select the **First-Come-First-Served (FCFS)** option in the SCHEDULER/Policies view
- Time slice should be considered as **seconds.**
- Create four processes P1, P2, P3 and P4 from source code respectively (Use the Priority drop-down list in the PROGRAM LIST / Process View): **3, 2, 4**,**1**
- Slide the Speed selector half-way down and then hit the START button.
- **Arrival delay** should be considered in **seconds** in the OS simulator

Now, give answer for the following:

a) What is the order in which processes are executed?

P1, P2, P3, P4

b) What is the ***Elapsed time , Average Process Waiting Time*** and ***Average Burst Period*** and of each process? (To see this, Click on VIEWS button available on the left of your OS control, the click VIEW LOG)

Process	***Arrival Time/Delay***	***Elapsed Time (sec)***	***Average Process Waiting Time (sec)***	***Average Burst Period***
P1	0	40.800	0.38 secs	95
P2	0	36.975	41.69 sec	79
P3	0	55.335	78.92 sec	135
P4	0	24.225	134.51 sec	55

```
00:17:33(32.915) [OS0: 00000000]: P1(4)  loaded into the READY queue [Memory=1 pages, Priority=3]
00:17:39(39.258) [OS0: 00000000]: P2(5)  loaded into the READY queue [Memory=1 pages, Priority=2]
00:17:46(45.566) [OS0: 00000000]: P3(6)  loaded into the READY queue [Memory=1 pages, Priority=4]
00:17:51(50.958) [OS0: 00000000]: P4(7)  loaded into the READY queue [Memory=1 pages, Priority=1]
00:17:55(54.645) [OS0: 00000000]:
00:17:55(54.645) [OS0: 00000000]: NEW SCHEDULER SESSION STARTED
00:17:55(54.645) [OS0: 00000000]: Scheduling scheme is First-Come-First-Served
00:17:55(54.645) [OS0: 00000000]: Memory allocation policy is First Fit
00:17:55(55.021) [OS0: 00000000]: P1(4)  moved from READY to RUNNING state on CPU 0
00:18:36(35.821) [OS0: 00000095]: P1(4)  terminated normally
00:18:36(35.821) [OS0: 00000095]: *** STATS FOR PROCESS P1(4) ***
00:18:36(35.821) [OS0: 00000095]: Elapsed Time = 00:00:41(40.800)
00:18:36(35.821) [OS0: 00000095]: Avg. Waiting Time = 0.38 sec
00:18:36(35.821) [OS0: 00000095]: Avg. Burst Period = 95
00:18:36(35.821) [OS0: 00000095]: Tick Count = 95
00:18:36(35.821) [OS0: 00000095]: Memory Swap Count = 0
00:18:36(36.331) [OS0: 00000096]: P2(5)  moved from READY to RUNNING state on CPU 0
00:19:13(13.306) [OS0: 00000175]: P2(5)  terminated normally
00:19:13(13.306) [OS0: 00000175]: *** STATS FOR PROCESS P2(5) ***
00:19:13(13.306) [OS0: 00000175]: Elapsed Time = 00:00:37(36.975)
00:19:13(13.306) [OS0: 00000175]: Avg. Waiting Time = 41.69 sec
00:19:13(13.306) [OS0: 00000175]: Avg. Burst Period = 79
00:19:13(13.306) [OS0: 00000175]: Tick Count = 79
00:19:13(13.306) [OS0: 00000175]: Memory Swap Count = 0
00:19:14(13.561) [OS0: 00000176]: P3(6)  moved from READY to RUNNING state on CPU 0
00:20:09(08.895) [OS0: 00000311]: P3(6)  terminated normally
00:20:09(08.895) [OS0: 00000311]: *** STATS FOR PROCESS P3(6) ***
00:20:09(08.895) [OS0: 00000311]: Elapsed Time = 00:00:55(55.335)
00:20:09(08.895) [OS0: 00000311]: Avg. Waiting Time = 78.92 sec
00:20:09(08.895) [OS0: 00000311]: Avg. Burst Period = 135
00:20:09(08.895) [OS0: 00000311]: Tick Count = 135
00:20:09(08.895) [OS0: 00000311]: Memory Swap Count = 0
00:20:09(09.151) [OS0: 00000312]: P4(7)  moved from READY to RUNNING state on CPU 0
00:20:33(33.376) [OS0: 00000367]: P4(7)  terminated normally
```

```
00:20:09(09.151) [OS0: 00000312]: P4(7)  moved from READY to RUNNING state on CPU 0
00:20:33(33.376) [OS0: 00000367]: P4(7)  terminated normally
00:20:33(33.376) [OS0: 00000367]: *** STATS FOR PROCESS P4(7) ***
00:20:33(33.376) [OS0: 00000367]: Elapsed Time = 00:00:24(24.225)
00:20:33(33.376) [OS0: 00000367]: Avg. Waiting Time = 134.51 sec
00:20:33(33.376) [OS0: 00000367]: Avg. Burst Period = 55
00:20:33(33.376) [OS0: 00000367]: Tick Count = 55
00:20:33(33.376) [OS0: 00000367]: Memory Swap Count = 0
00:20:34(33.570) [OS0: 00000368]: Avg. Process Waiting Time = 63.87 sec
```

PART-II_B

- Select the **Shortest Job First (SJF)** option in the SCHEDULER/Policies view
- Select the Priority (static) as **Pre-emptive** option in the SCHEDULER/Policies view
- Time slice should be considered as **seconds.**
- Create four processes P1, P2, P3 and P4 from source codes respectively (Use the Priority drop-down list in the PROGRAM LIST / Process View): **3, 2, 4**,**1**
- Slide the Speed selector half-way down and then hit the START button.
- **Arrival delay** should be considered in **seconds** in the OS simulator

Now, give answer for the following:

a) What is the order in which processes are executed?

P1, P2, P3, P4

b) What is the ***Elapsed time , Average Process Waiting Time*** and ***Average Burst Period*** and of each process? (To see this, Click on VIEWS button available on the left of your OS control, the click VIEW LOG)

Process	***Arrival Time/Delay***		***Elapsed Time (sec)***	***Average Process Waiting Time (sec)***	***Average Burst Period***
P1	0		39.781	0.36	95
P2	0		33.660	40.65	79
P3	0		55.080	74.56	135
P4	0		24.480	129.9	55

```
02:24:11(11.142) [OS0: 00000000]: NEW SCHEDULER SESSION STARTED
02:24:11(11.142) [OS0: 00000000]: Scheduling scheme is Shortest-Job-First with Pre-emptive priority
02:24:11(11.142) [OS0: 00000000]: Memory allocation policy is First Fit
02:24:11(11.497) [OS0: 00000000]: P1(4)  moved from READY to RUNNING state on CPU 0
02:24:51(51.277) [OS0: 00000095]: P1(4)  terminated normally
02:24:51(51.277) [OS0: 00000095]: *** STATS FOR PROCESS P1(4) ***
02:24:51(51.277) [OS0: 00000095]: Elapsed Time = 00:00:40(39.781)
02:24:51(51.277) [OS0: 00000095]: Avg. Waiting Time = 0.36 sec
02:24:51(51.277) [OS0: 00000095]: Avg. Burst Period = 95
02:24:51(51.277) [OS0: 00000095]: Tick Count = 95
02:24:51(51.277) [OS0: 00000095]: Memory Swap Count = 0
02:24:52(51.787) [OS0: 00000096]: P2(5)  moved from READY to RUNNING state on CPU 0
02:25:25(25.446) [OS0: 00000175]: P2(5)  terminated normally
02:25:25(25.446) [OS0: 00000175]: *** STATS FOR PROCESS P2(5) ***
02:25:25(25.446) [OS0: 00000175]: Elapsed Time = 00:00:34(33.660)
02:25:25(25.446) [OS0: 00000175]: Avg. Waiting Time = 40.65 sec
02:25:25(25.446) [OS0: 00000175]: Avg. Burst Period = 79
02:25:25(25.446) [OS0: 00000175]: Tick Count = 79
02:25:25(25.446) [OS0: 00000175]: Memory Swap Count = 0
02:25:26(25.702) [OS0: 00000176]: P3(6)  moved from READY to RUNNING state on CPU 0
02:26:21(20.781) [OS0: 00000311]: P3(6)  terminated normally
02:26:21(20.781) [OS0: 00000311]: *** STATS FOR PROCESS P3(6) ***
02:26:21(20.781) [OS0: 00000311]: Elapsed Time = 00:00:55(55.080)
02:26:21(20.781) [OS0: 00000311]: Avg. Waiting Time = 74.56 sec
02:26:21(20.781) [OS0: 00000311]: Avg. Burst Period = 135
02:26:21(20.781) [OS0: 00000311]: Tick Count = 135
02:26:21(20.781) [OS0: 00000311]: Memory Swap Count = 0
02:26:21(21.037) [OS0: 00000312]: P4(7)  moved from READY to RUNNING state on CPU 0
02:26:46(45.517) [OS0: 00000367]: P4(7)  terminated normally
02:26:46(45.517) [OS0: 00000367]: *** STATS FOR PROCESS P4(7) ***
02:26:46(45.517) [OS0: 00000367]: Elapsed Time = 00:00:24(24.480)
02:26:46(45.517) [OS0: 00000367]: Avg. Waiting Time = 129.9 sec
02:26:46(45.517) [OS0: 00000367]: Avg. Burst Period = 55
02:26:46(45.517) [OS0: 00000367]: Tick Count = 55
```

```
02:26:46(45.517) [OS0: 00000367]: P4(7)  terminated normally
02:26:46(45.517) [OS0: 00000367]: *** STATS FOR PROCESS P4(7) ***
02:26:46(45.517) [OS0: 00000367]: Elapsed Time = 00:00:24(24.480)
02:26:46(45.517) [OS0: 00000367]: Avg. Waiting Time = 129.9 sec
02:26:46(45.517) [OS0: 00000367]: Avg. Burst Period = 55
02:26:46(45.517) [OS0: 00000367]: Tick Count = 55
02:26:46(45.517) [OS0: 00000367]: Memory Swap Count = 0
02:26:46(45.704) [OS0: 00000368]: Avg. Process Waiting Time = 61.36 sec
```

PART-II_C

- Select the **Round Robin (RR) with 5 seconds as time slice** option in the SCHEDULER/Policies view.
- Select the Priority (static) as **Pre-emptive** option in the SCHEDULER/Policies view
- Time slice should be taken in terms of **seconds** instead of **ticks**
- Create four processes P1, P2, P3 and P4 from source codes respectively (Use the Priority drop-down list in the PROGRAM LIST / Process View): **3, 2, 4**,**1**
- Slide the Speed selector half-way down and then hit the START button.
- **Arrival delay** should be considered in **seconds** in the OS simulator

Now, give answer for the following:

a) What is the order in which processes are executed?

P4, P2, P3,P1

b) What is the ***Elapsed time , Average Process Waiting Time*** and ***Average Burst Period*** and of each process? (To see this, Click on VIEWS button available on the left of your OS control, the click VIEW LOG)

Process	***Arrival Time/Delay***	***Elapsed Time (sec)***	***Average Process Waiting Time (sec)***	***Average Burst Period***
P4	0	91	8.65 sec	6
P2	0	120.05	6.64 sec	6
P3	0	120.50	6.08 sec	7
P1	0	180.38	88.02 sec	47

```
02:38:09(09.208) [OS0: 00000000]:
02:38:09(09.208) [OS0: 00000000]: NEW SCHEDULER SESSION STARTED
02:38:09(09.208) [OS0: 00000000]: Scheduling scheme is Round Robin using Static Priority (Pre-emptive): Time slots 5 secs
02:38:09(09.208) [OS0: 00000000]: Memory allocation policy is First Fit
02:38:10(09.546) [OS0: 00000000]: P1(4)  moved from READY to RUNNING state on CPU 0
02:38:15(14.646) [OS0: 00000009]: P1(4)  moved from RUNNING to READY state
02:38:15(14.902) [OS0: 00000009]: P2(5)  moved from READY to RUNNING state on CPU 0
02:38:19(19.491) [OS0: 00000016]: P2(5)  moved from RUNNING to READY state
02:38:20(20.002) [OS0: 00000016]: P3(6)  moved from READY to RUNNING state on CPU 0
02:38:25(24.592) [OS0: 00000023]: P3(6)  moved from RUNNING to READY state
02:38:25(24.847) [OS0: 00000023]: P4(7)  moved from READY to RUNNING state on CPU 0
02:38:29(29.437) [OS0: 00000030]: P4(7)  moved from RUNNING to READY state
02:38:30(29.691) [OS0: 00000030]: P2(5)  moved from READY to RUNNING state on CPU 0
02:38:35(35.047) [OS0: 00000037]: P2(5)  moved from RUNNING to READY state
02:38:35(35.301) [OS0: 00000037]: P3(6)  moved from READY to RUNNING state on CPU 0
02:38:40(39.637) [OS0: 00000043]: P3(6)  moved from RUNNING to READY state
02:38:40(39.891) [OS0: 00000043]: P3(6)  moved from READY to RUNNING state on CPU 0
02:38:44(44.481) [OS0: 00000052]: P3(6)  moved from RUNNING to READY state
02:38:45(44.736) [OS0: 00000052]: P4(7)  moved from READY to RUNNING state on CPU 0
02:38:50(50.091) [OS0: 00000059]: P4(7)  moved from RUNNING to READY state
02:38:50(50.347) [OS0: 00000059]: P2(5)  moved from READY to RUNNING state on CPU 0
02:38:54(54.427) [OS0: 00000066]: P2(5)  moved from RUNNING to READY state
02:38:55(54.682) [OS0: 00000066]: P3(6)  moved from READY to RUNNING state on CPU 0
02:39:00(00.291) [OS0: 00000074]: P3(6)  moved from RUNNING to READY state
02:39:01(00.547) [OS0: 00000074]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:05(04.627) [OS0: 00000081]: P4(7)  moved from RUNNING to READY state
02:39:05(04.882) [OS0: 00000081]: P2(5)  moved from READY to RUNNING state on CPU 0
02:39:09(09.472) [OS0: 00000088]: P2(5)  moved from RUNNING to READY state
02:39:10(09.726) [OS0: 00000088]: P3(6)  moved from READY to RUNNING state on CPU 0
02:39:15(15.337) [OS0: 00000096]: P3(6)  moved from RUNNING to READY state
02:39:16(15.591) [OS0: 00000096]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:20(19.927) [OS0: 00000102]: P4(7)  moved from RUNNING to READY state
02:39:20(20.181) [OS0: 00000102]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:25(24.516) [OS0: 00000110]: P4(7)  moved from RUNNING to READY state
```

```
02:39:20(20.181)  [OS0:  00000102]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:25(24.516)  [OS0:  00000110]: P4(7)  moved from RUNNING to READY state
02:39:25(24.772)  [OS0:  00000110]: P2(5)  moved from READY to RUNNING state on CPU 0
02:39:30(29.617)  [OS0:  00000115]: P2(5)  moved from RUNNING to READY state
02:39:30(29.871)  [OS0:  00000115]: P2(5)  moved from READY to RUNNING state on CPU 0
02:39:34(34.461)  [OS0:  00000124]: P2(5)  moved from RUNNING to READY state
02:39:35(34.716)  [OS0:  00000124]: P3(6)  moved from READY to RUNNING state on CPU 0
02:39:40(40.326)  [OS0:  00000132]: P3(6)  moved from RUNNING to READY state
02:39:41(40.581)  [OS0:  00000132]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:45(45.171)  [OS0:  00000136]: P4(7)  moved from RUNNING to READY state
02:39:45(45.427)  [OS0:  00000136]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:50(49.761)  [OS0:  00000144]: P4(7)  moved from RUNNING to READY state
02:39:50(50.017)  [OS0:  00000144]: P4(7)  moved from READY to RUNNING state on CPU 0
02:39:55(55.372)  [OS0:  00000152]: P4(7)  terminated normally
02:39:55(55.372)  [OS0:  00000152]: *** STATS FOR PROCESS P4(7) ***
02:39:55(55.372)  [OS0:  00000152]: Elapsed Time = 00:01:31(30.525)
02:39:55(55.372)  [OS0:  00000152]: Avg. Waiting Time = 8.65 sec
02:39:55(55.372)  [OS0:  00000152]: Avg. Burst Period = 6
02:39:55(55.372)  [OS0:  00000152]: Tick Count = 55
02:39:55(55.372)  [OS0:  00000152]: Memory Swap Count = 0
02:39:56(55.627)  [OS0:  00000153]: P2(5)  moved from READY to RUNNING state on CPU 0
02:39:58(57.666)  [OS0:  00000154]: P2(5)  moved from RUNNING to READY state
02:39:58(57.921)  [OS0:  00000154]: P2(5)  moved from READY to RUNNING state on CPU 0
02:39:60(59.706)  [OS0:  00000156]: P2(5)  moved from RUNNING to READY state
02:39:60(59.962)  [OS0:  00000156]: P2(5)  moved from READY to RUNNING state on CPU 0
02:40:05(04.807)  [OS0:  00000165]: P2(5)  moved from RUNNING to READY state
02:40:05(05.061)  [OS0:  00000165]: P2(5)  moved from READY to RUNNING state on CPU 0
02:40:10(09.652)  [OS0:  00000174]: P2(5)  moved from RUNNING to READY state
02:40:10(09.906)  [OS0:  00000174]: P2(5)  moved from READY to RUNNING state on CPU 0
02:40:14(14.497)  [OS0:  00000183]: P2(5)  moved from RUNNING to READY state
02:40:15(14.752)  [OS0:  00000183]: P2(5)  moved from READY to RUNNING state on CPU 0
02:40:20(19.852)  [OS0:  00000190]: P2(5)  terminated normally
```

```
02:40:20(19.852)  [OS0:  00000190]: P2(5)  terminated normally
02:40:20(19.852)  [OS0:  00000190]: *** STATS FOR PROCESS P2(5) ***
02:40:20(19.852)  [OS0:  00000190]: Elapsed Time = 00:02:05(04.950)
02:40:20(19.852)  [OS0:  00000190]: Avg. Waiting Time = 6.64 sec
02:40:20(19.852)  [OS0:  00000190]: Avg. Burst Period = 6
02:40:20(19.852)  [OS0:  00000190]: Tick Count = 79
02:40:20(19.852)  [OS0:  00000190]: Memory Swap Count = 0
02:40:20(20.362)  [OS0:  00000191]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:22(22.401)  [OS0:  00000192]: P3(6)  moved from RUNNING to READY state
02:40:23(22.657)  [OS0:  00000192]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:25(24.696)  [OS0:  00000195]: P3(6)  moved from RUNNING to READY state
02:40:25(24.951)  [OS0:  00000195]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:30(29.541)  [OS0:  00000204]: P3(6)  moved from RUNNING to READY state
02:40:30(29.797)  [OS0:  00000204]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:35(34.641)  [OS0:  00000214]: P3(6)  moved from RUNNING to READY state
02:40:35(34.896)  [OS0:  00000214]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:40(39.742)  [OS0:  00000224]: P3(6)  moved from RUNNING to READY state
02:40:40(39.997)  [OS0:  00000224]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:45(44.586)  [OS0:  00000233]: P3(6)  moved from RUNNING to READY state
02:40:45(44.842)  [OS0:  00000233]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:50(49.686)  [OS0:  00000244]: P3(6)  moved from RUNNING to READY state
02:40:50(49.942)  [OS0:  00000244]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:56(55.551)  [OS0:  00000255]: P3(6)  moved from RUNNING to READY state
02:40:56(55.807)  [OS0:  00000255]: P3(6)  moved from READY to RUNNING state on CPU 0
02:40:60(59.632)  [OS0:  00000263]: P3(6)  moved from RUNNING to READY state
02:40:60(59.887)  [OS0:  00000263]: P3(6)  moved from READY to RUNNING state on CPU 0
02:41:05(04.732)  [OS0:  00000273]: P3(6)  moved from RUNNING to READY state
02:41:05(04.986)  [OS0:  00000273]: P3(6)  moved from READY to RUNNING state on CPU 0
02:41:10(10.086)  [OS0:  00000280]: P3(6)  terminated normally
02:41:10(10.086)  [OS0:  00000280]: *** STATS FOR PROCESS P3(6) ***
02:41:10(10.086)  [OS0:  00000280]: Elapsed Time = 00:02:50(50.085)
02:41:10(10.086)  [OS0:  00000280]: Avg. Waiting Time = 6.08 sec
02:41:10(10.086)  [OS0:  00000280]: Avg. Burst Period = 7
02:41:10(10.086)  [OS0:  00000280]: Tick Count = 135
```

```
02:41:10(10.086)  [OS0:  00000280]: P3(6)  terminated normally
02:41:10(10.086)  [OS0:  00000280]: *** STATS FOR PROCESS P3(6) ***
02:41:10(10.086)  [OS0:  00000280]: Elapsed Time = 00:02:50(50.085)
02:41:10(10.086)  [OS0:  00000280]: Avg. Waiting Time = 6.08 sec
02:41:10(10.086)  [OS0:  00000280]: Avg. Burst Period = 7
02:41:10(10.086)  [OS0:  00000280]: Tick Count = 135
02:41:10(10.086)  [OS0:  00000280]: Memory Swap Count = 0
02:41:10(10.342)  [OS0:  00000281]: P1(4)  moved from READY to RUNNING state on CPU 0
02:41:47(47.316)  [OS0:  00000367]: P1(4)  terminated normally
02:41:47(47.316)  [OS0:  00000367]: *** STATS FOR PROCESS P1(4) ***
02:41:47(47.316)  [OS0:  00000367]: Elapsed Time = 00:03:38(37.770)
02:41:47(47.316)  [OS0:  00000367]: Avg. Waiting Time = 88.02 sec
02:41:47(47.316)  [OS0:  00000367]: Avg. Burst Period = 47
02:41:47(47.316)  [OS0:  00000367]: Tick Count = 95
02:41:47(47.316)  [OS0:  00000367]: Memory Swap Count = 0
02:41:48(47.513)  [OS0:  00000368]: Avg. Process Waiting Time = 65.89 sec
```

PART-II_D

a) Plot a graph from the results obtained by FCFS, SJF and Round Robin scheduling and explain which algorithm is better among these with proper justification

P1 = Priority3, P2 = Priority2, P3 = Priority4, P4 = Priority1

Elapse Time:

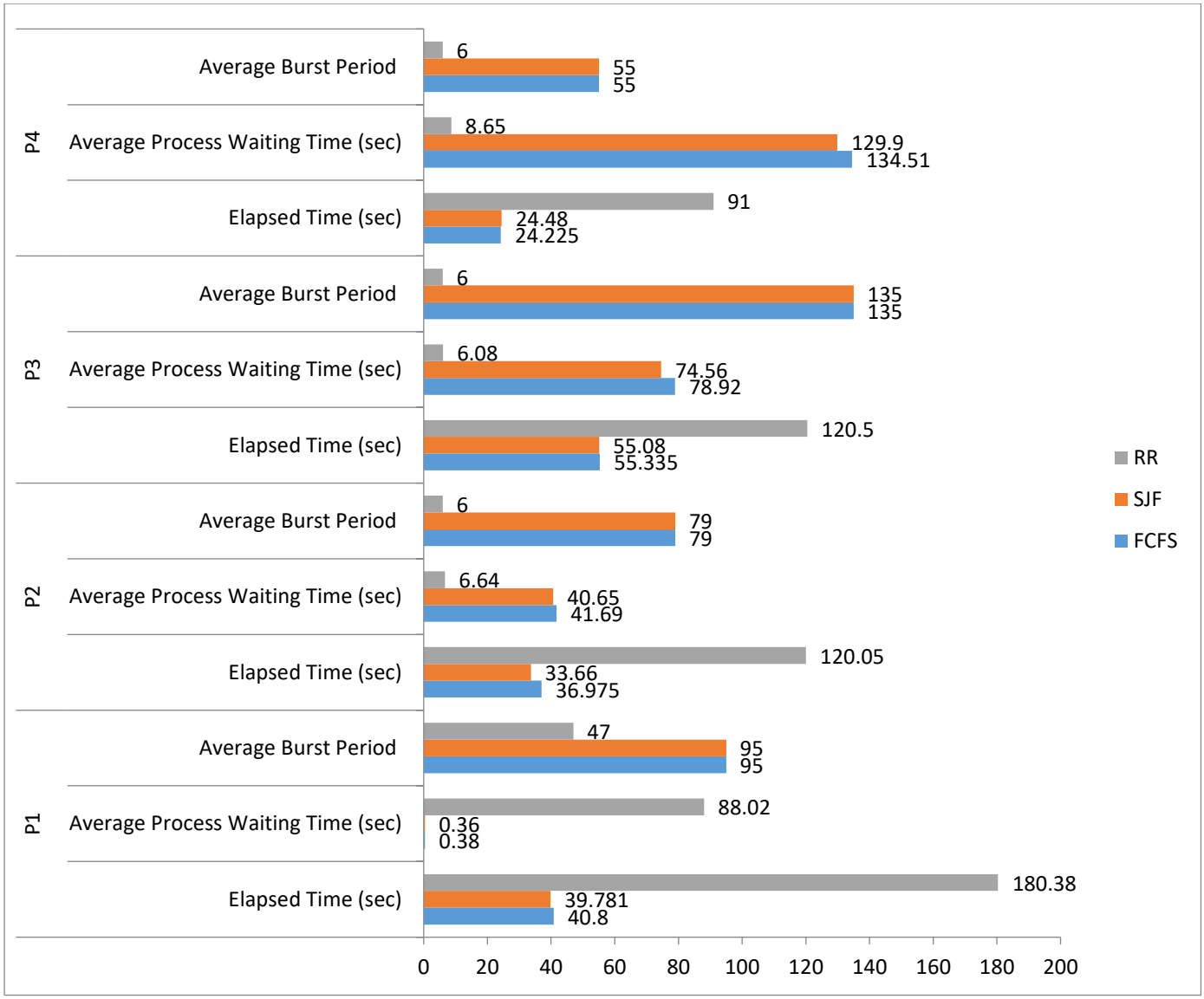
1. Elapsed time e.g. waiting for I/O operations is higher for Round Robin algorithm, compared to FCFS and SJF algorithm. Round Robin allocates fixed slice time and scheduler keep the ready queue as FIFO queue of processes and so other processes in the queue have higher elapse time.
2. Priority of the process does not have impact on the elapsed time

Average Process Waiting time:

1. Average Process waiting time is approx. similar for FCFS and SJF scheduler for the given program. However, for Priority1 process average waiting time for Round Robin pre-emptive scheduling is very less than FCFS and SJF.
2. SJF is optimal in terms of average waiting time for a given set of processes

Average Burst Period:

1. Round Robin scheduler has lower average burst period as compared to FCFS and SJF scheduling algorithm.
2. Burst Period for priority1 process has improved performance in FCFS and SJF compared to lower priority processes. Performance increases as the priority of process changes from lower to higher.


P4
Average Burst Period
6
55
55
Average Process Waiting Time (sec)
8.65
129.9
134.51
Elapsed Time (sec)
91
24.48
24.225
P3
Average Burst Period
6
135
135
Average Process Waiting Time (sec)
6.08
74.56
78.92
Elapsed Time (sec)
120.5
55.08
55.335
P2
Average Burst Period
6
79
79
Average Process Waiting Time (sec)
6.64
40.65
41.69
Elapsed Time (sec)
120.05
33.66
36.975
P1
Average Burst Period
47
95
95
Average Process Waiting Time (sec)
88.02
0.36
0.38
Elapsed Time (sec)
180.38
39.781
40.8
0
20
40
60
80
100
120
140
160
180
200
RR
SJF
FCFS

Part III: Multi-Threading

Consider the following source code

```
program ThreadTest
        total = 100
        sub thread1 as thread
            for i = 1 to 2
                total = total - i

            next
        end sub
        sub thread2 as thread
            for i = 3 to 4
                total = total + i
                call thread1
            next
        end sub
        call thread2
        wait
        writeln ("Total =", total)
end
```

Compile the above source code and load it in the main memory. Create a single process, choose RR scheduling algorithm with time quantum of 5 seconds. Run the Process.

Answer the following questions:

a) What is the value of "Total" ? – **7**

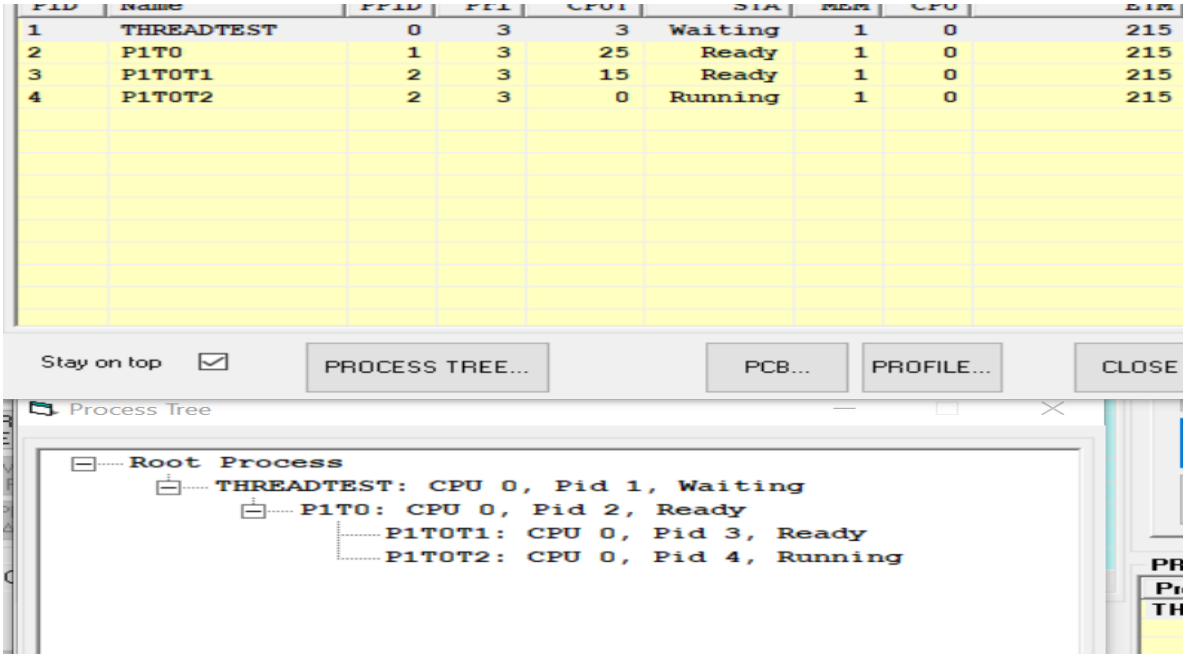
b) How many processes and how many threads are created? **1 Process and 3 Threads**

c) Identify the name of the processes and threads. **Process name - P1, Thread- P1T0, P1T0T1, P1T0T2**

d) What is the PID and PPID of the processes and threads created?

PID	Name	PPID
1	THREADTEST	0
2	P1T0	1
3	P1T0T1	2
4	P1T0T2	2

e) Represent the parent and child relationship using tree representation



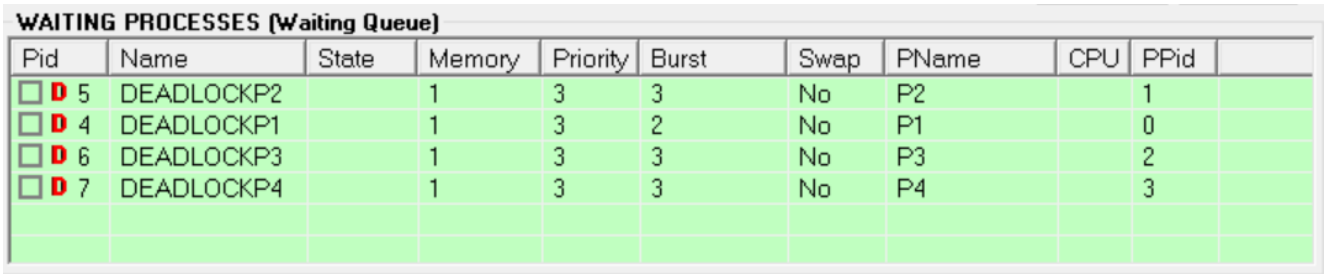
PART-IV:Deadlock

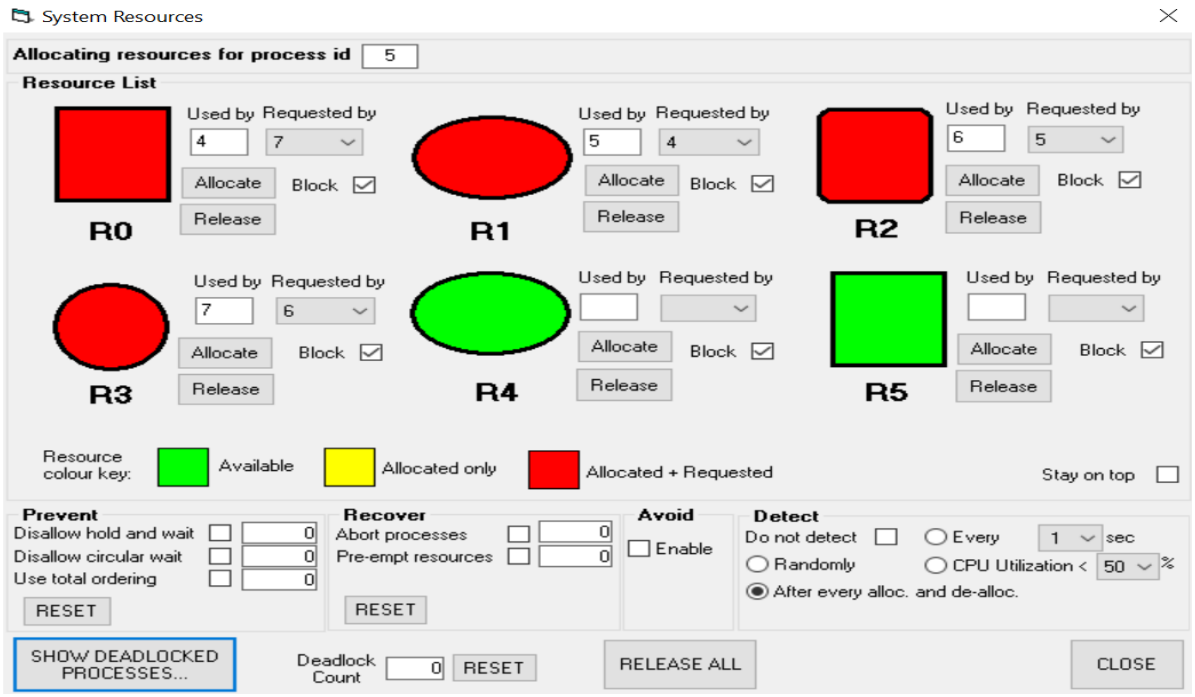
```
program DeadlockP1
        resource(1, allocate)
        wait(3)
        resource(2, allocate)
        for n = 1 to 20
        next
end

program DeadlockP2
        resource(2, allocate)
        wait(3)
        resource(3, allocate)
        for n = 1 to 20
        next
end

program DeadlockP3
        resource(3, allocate)
        wait(3)
        resource(4, allocate)
        for n = 1 to 20
        next
end

program DeadlockP4
        resource(4, allocate)
        wait(3)
        resource(5, allocate)
        for n = 1 to 20
        next
end

program DeadlockP5
        resource(5, allocate)
        wait(3)
        resource(1, allocate)
        for n = 1 to 20
        next
end
```

- The above source code creates a program which attempts to allocate **two** resources for itself.
- After the first allocation it waits for 3 seconds and tries to allocate another resource.
- Finally, it counts from 1 to 20 in a loop and then terminates

Problem Statement:

- Create Four processes from the above source code .
- Allocate the resources for each process as below:
  - P1→ R0 and R1
  - P2→ R1 and R2
  - P3→ R2 and R3
  - P4→ R3 and R0
- Save each program into local directory
- Compile each one of the four-source code.
- Load in memory the four pieces of code generated.
- Now switch to the OS simulator.
- Create a single instance of each of the programs. You can do this by double-clicking on each of the program names in the PROGRAM LIST frame under the Program Name column.
- In the SCHEDULER frame select Round Robin (RR) scheduling policy with no priority in the Policies tab.
- Select the Views tab and click on the VIEW RESOURCES… button.
- In the OS Control tab use the START button to start the OS scheduler and observe the changing process states for few seconds.





1. How to avoid the deadlock by releasing either resource or a process (Justify your answer wrt cost effective)

There are two approaches for recover from deadlock – by A. Process termination or by B. Resource pre-emption

A. **Process Termination** – 1. *Abort all the deadlock processes* is of great expenses. The deadlock processes might have computed for long time and the result of these processes would be lost and require recalculation again. This would lead to huge computational cost.
Abort processes until deadlock is released – This approach performs abort one deadlocked process at a time until deadlock cycle is eliminated. There is considerable overhead to implement deadlock detection algorithm and eliminate the deadlocked processes.

B. **Resource pre-emption** – This method allows to pre-empt some resources from processes and allocate these resources to other process until the deadlock cycle is broken. This is cost effective method as resources are temporarily allotted to the priority or identified processes, once this resource is released by the process it is allocated to suspended process. This allows to avoid terminating the processes.

2. Draw a Resource Allocation Graph.