Process Management

Group 30

WHY PROCESS MANAGEMENT ?

Problem Statement

- The computer system has N processes that are to be run on the processor in order to complete the tasks.
- As all the processes cannot run simultaneously on the processor, there arises a need of a mechanism through which processes can be managed effectively.

INTRODUCTION

Scope of the project

- The computer system has only one processor.
- Limited size of the ready Queue.

Following has been Implemented

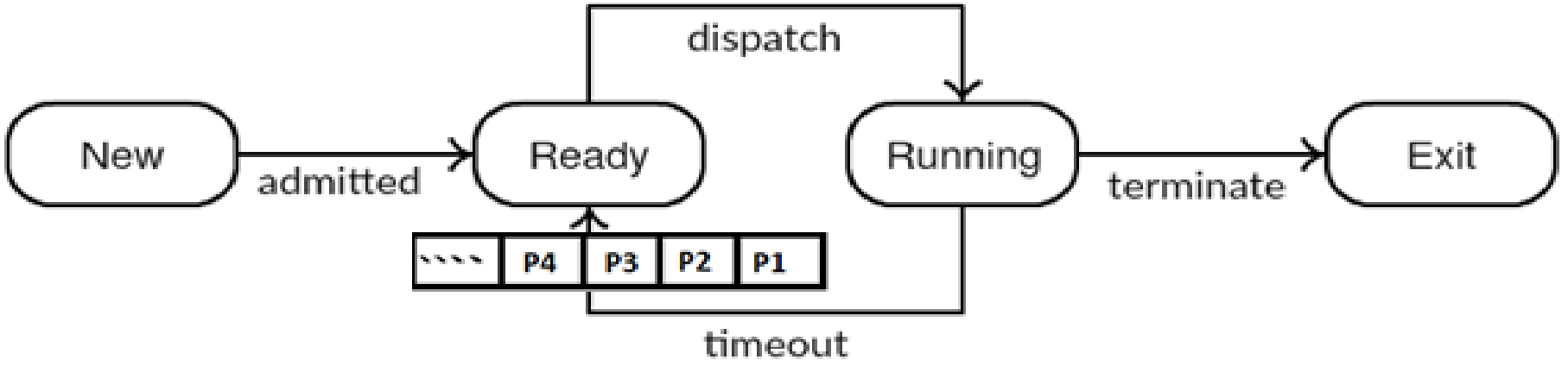
- IPC between Scheduler and Dispatcher using named pipes.
- Dynamic submission of processes - Variable number of processes can be submitted for execution at any time.
- Dynamic Data Memory - Dispatcher finds empty patches and memory is allocated to new processes accordingly.
- Signal Handling - Signal immunity for scheduler and dispatcher.

Contd…

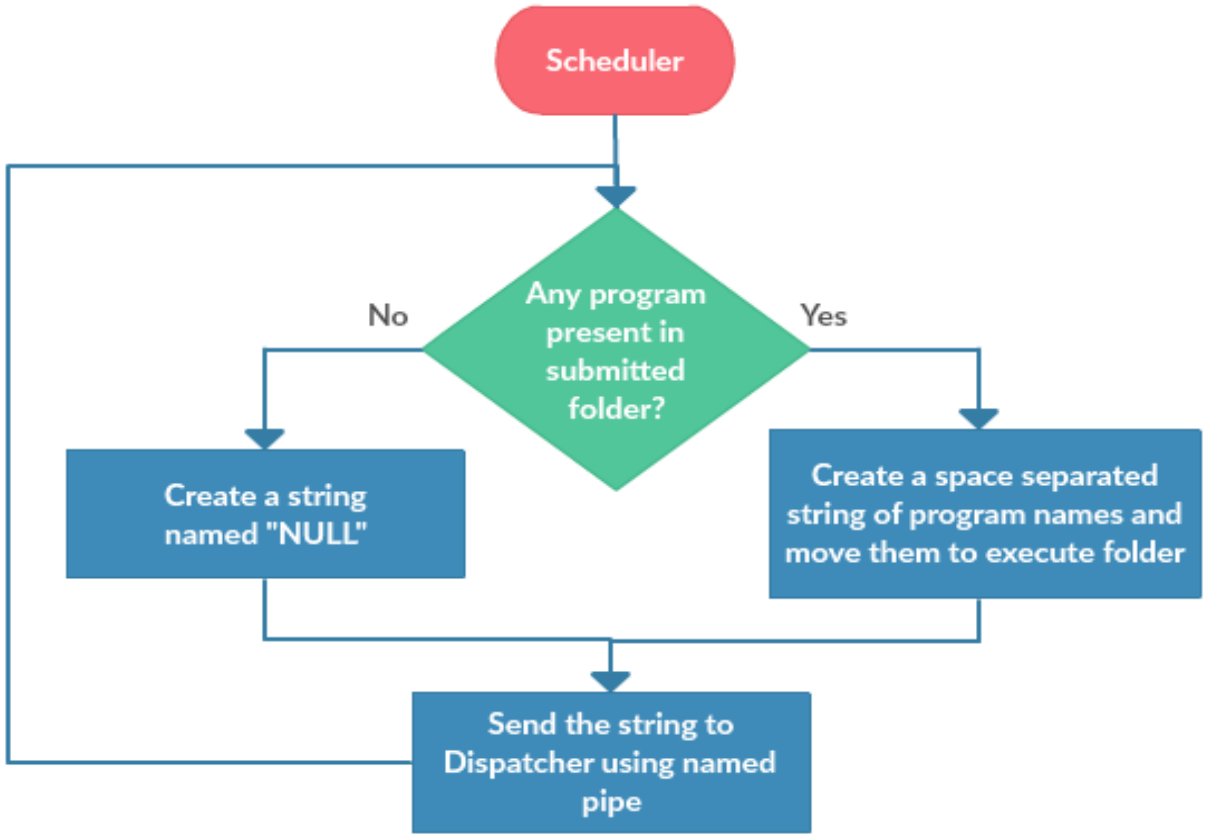
- History of processes - Time taken by process is saved in history file along with timestamp
- Log file generation - A log file is continuously updated which contains time stamp, PID, process name, Number of active processes, CPU efficiency
- CPU efficiency - CPU efficiency is computed after every time slice
- Proc folder - PCB or the process image is maintained for every process in separate files inside proc folder
- Turnaround time, response time and actual execution time of a process

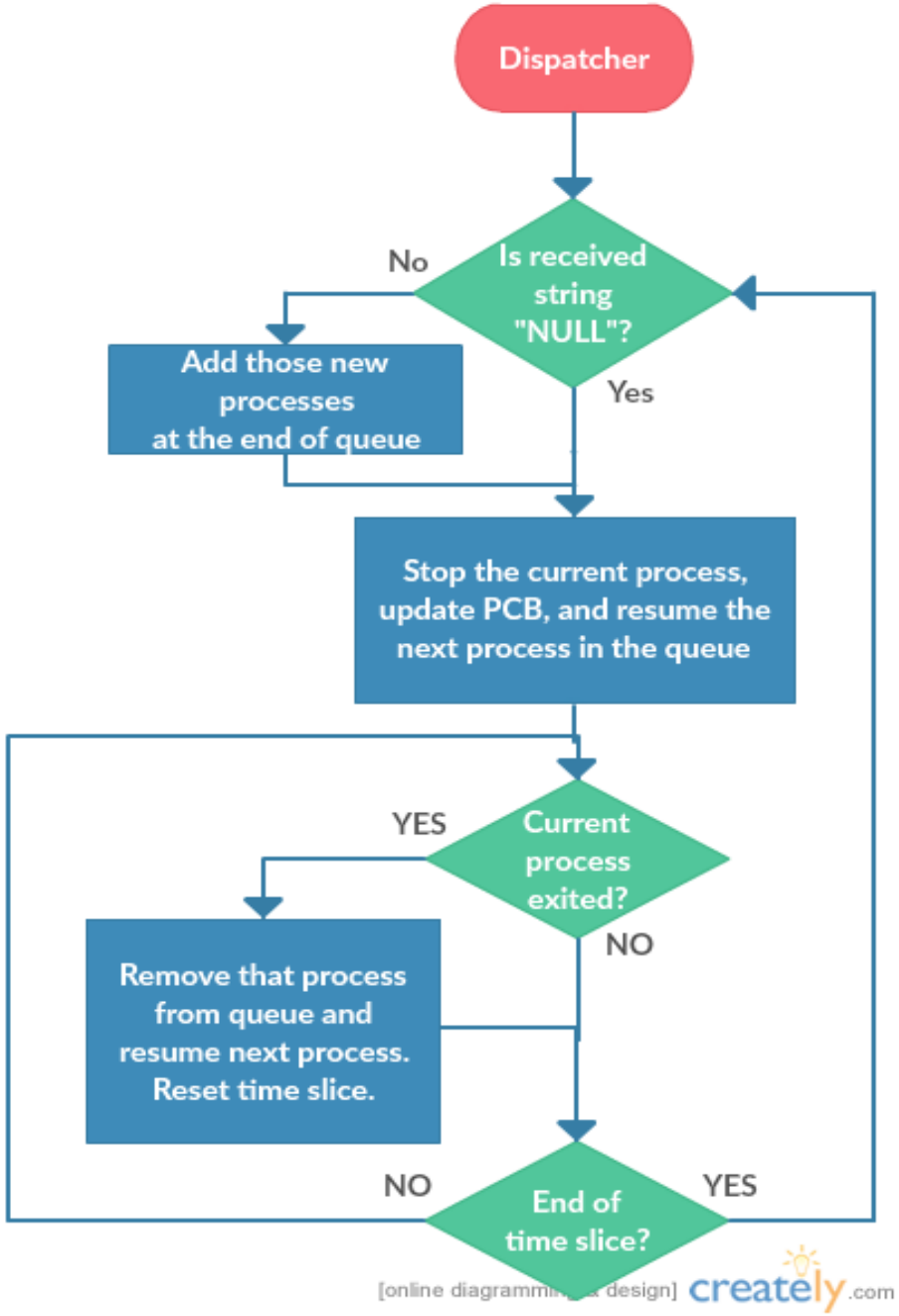
IMPLEMENTATION

State diagram



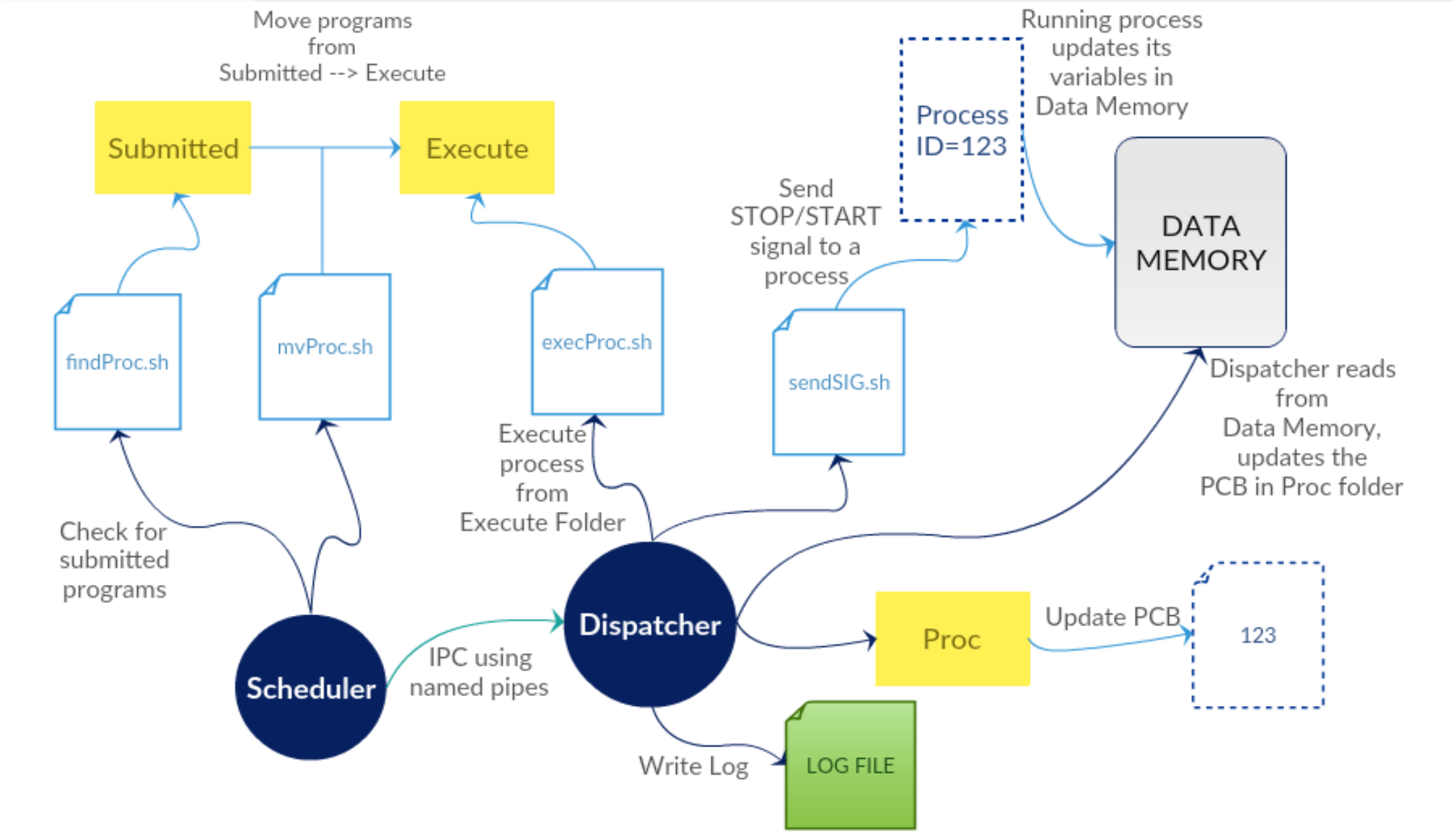
Flow Diagram





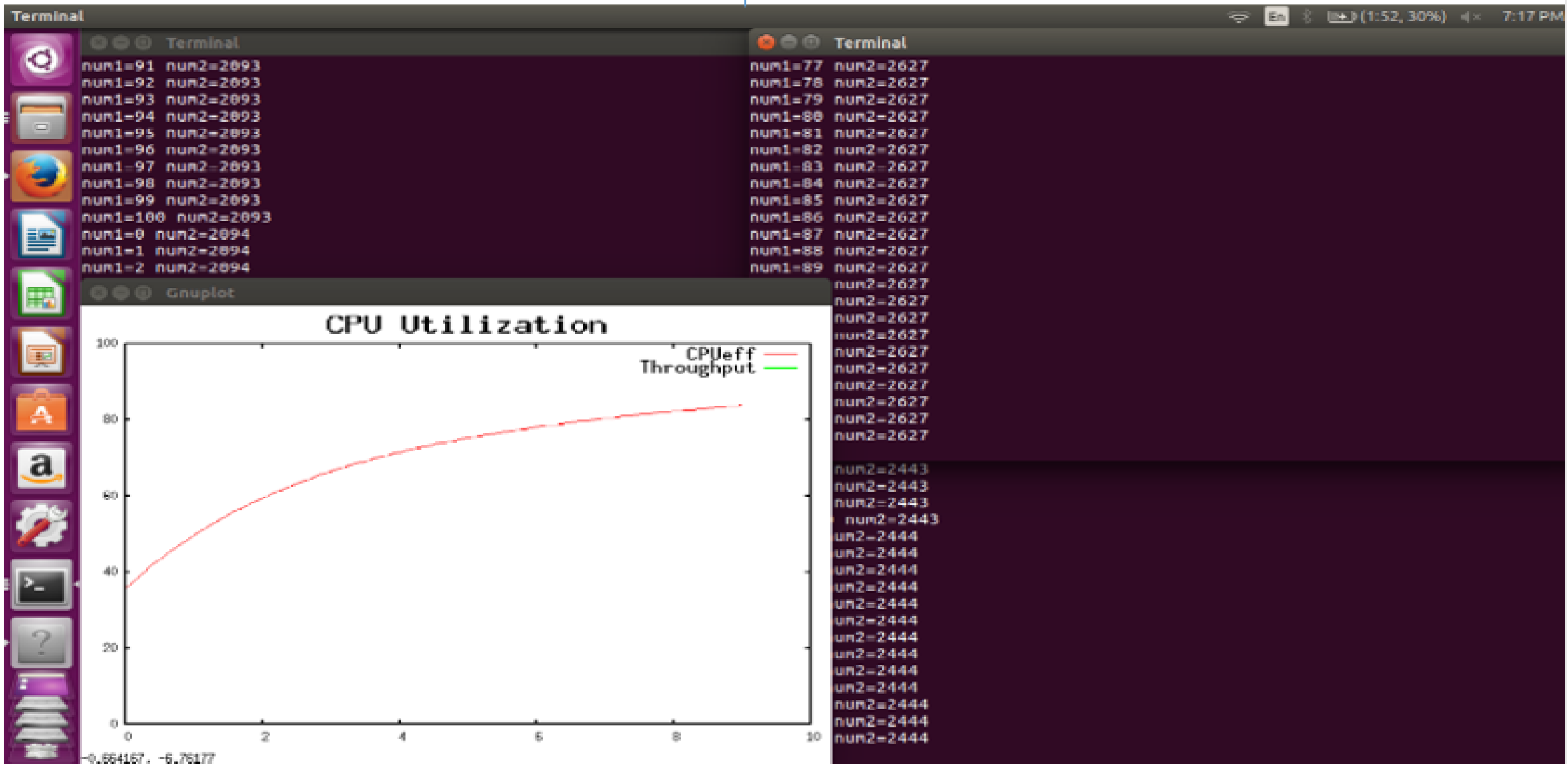
Flow diagram

Connection Diagram



Connection diagram

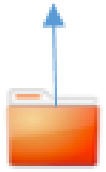
Screen shot of running project



Dispatcher & Scheduler

manager

Works as PCB block

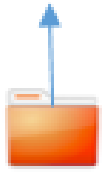


proc

Processes that are in ready queue

process_c_out_ex

New Processes

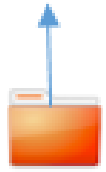


process_c_out_ submitted

Different C Prog

process_c_script

All required shell files



sh_script

DataMemory

Works as RAM

history

Process Turnaround time, Execution time & Waiting time are stored here

initiate_it

Click here to See the Magic

log

Maintain the log of all executed & executing processes

plotEff

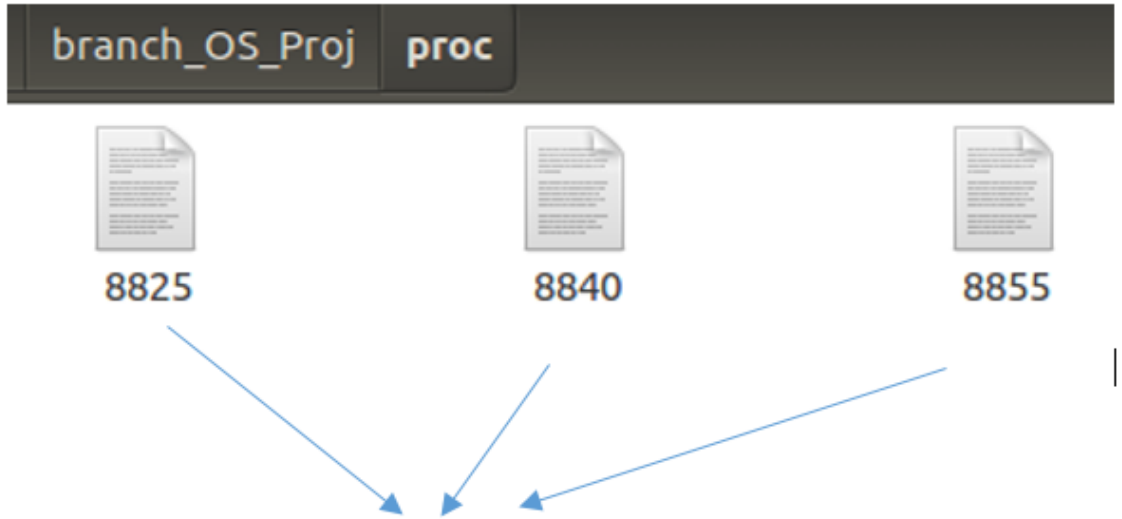
Maintain the data of CPU efficiency & Throughput

```
 1 ID=8825
 2 num2=675
 3 num1=65
 4 END
 5
 6
 7
 8
 9
10
11 ID=8840
12 num2=704
13 num1=82
14 END
15
16
17
18
19
20
21 ID=8855
22 num2=640
23 num1=83
24 END
25
26
27
28
29
30
31
```

Reserved memory area for one process

ID of process

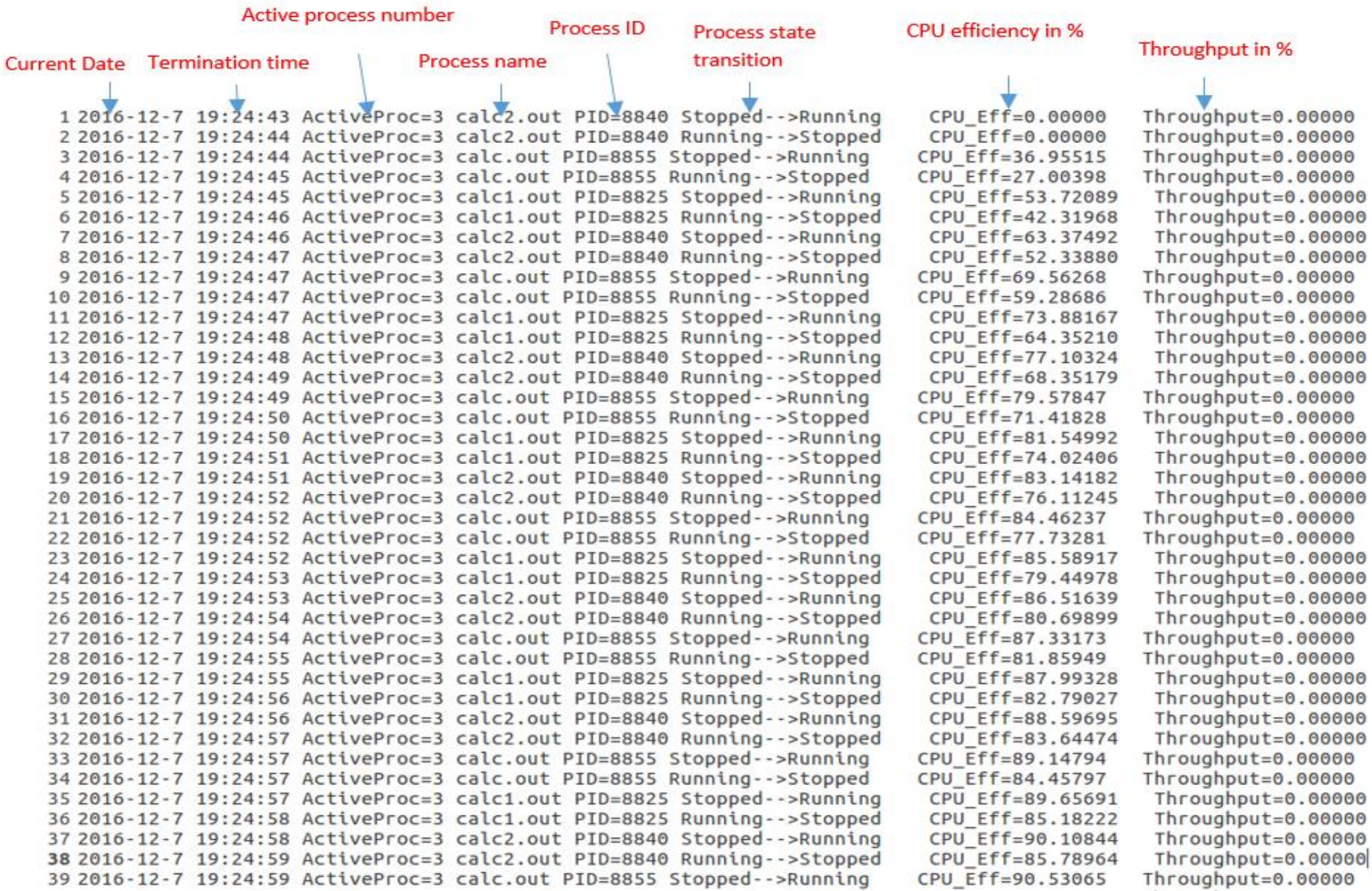
Values of the variables



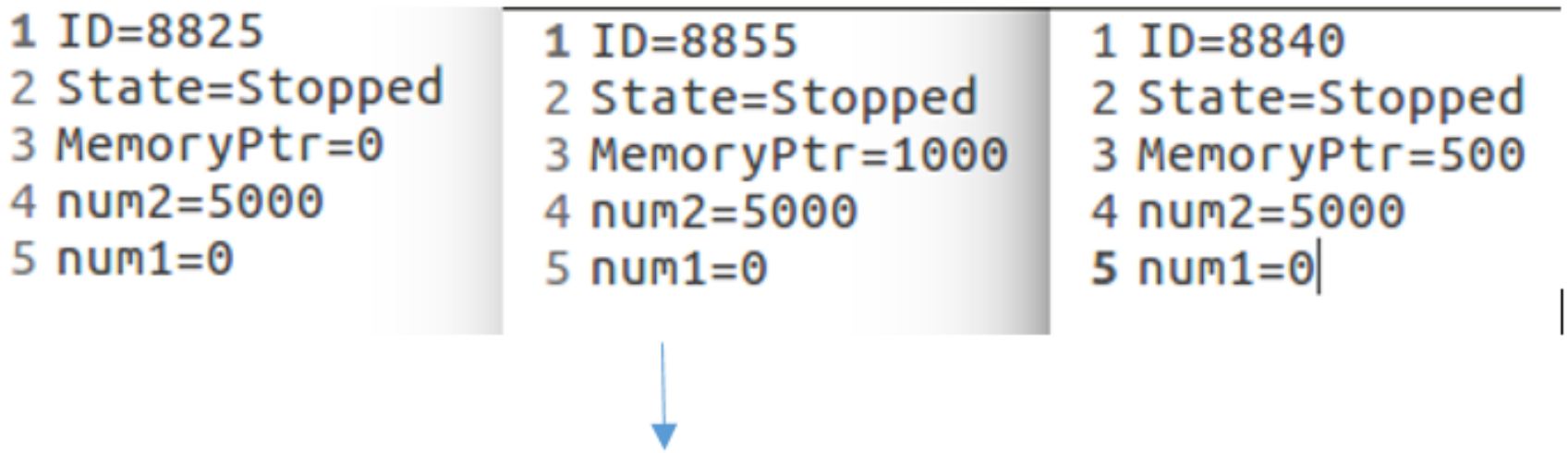
proc folder contains PCB of all the executed or executing processes as name of its ID


```
40 2016-12-7 19:25:0 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=86.42258   Throughput=0.00000
41 2016-12-7 19:25:0 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=90.90638   Throughput=0.00000
42 2016-12-7 19:25:1 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=86.97126   Throughput=0.00000
43 2016-12-7 19:25:1 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=91.24596   Throughput=0.00000
44 2016-12-7 19:25:2 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=87.46560   Throughput=0.00000
45 2016-12-7 19:25:2 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=91.56041   Throughput=0.00000
46 2016-12-7 19:25:2 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=87.88123   Throughput=0.00000
47 2016-12-7 19:25:2 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=91.83392   Throughput=0.00000
48 2016-12-7 19:25:3 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=88.31996   Throughput=0.00000
49 2016-12-7 19:25:3 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=92.10035   Throughput=0.00000
50 2016-12-7 19:25:4 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=88.72394   Throughput=0.00000
51 2016-12-7 19:25:4 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=92.34676   Throughput=0.00000
52 2016-12-7 19:25:5 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=89.07497   Throughput=0.00000
53 2016-12-7 19:25:5 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=92.56980   Throughput=0.00000
54 2016-12-7 19:25:6 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=89.41274   Throughput=0.00000
55 2016-12-7 19:25:6 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=92.77921   Throughput=0.00000
56 2016-12-7 19:25:6 ActiveProc=3 calc2.out PID=8840 Running-->Exited    CPU_Eff=92.72738   Throughput=4.19540
57 2016-12-7 19:25:6 ActiveProc=2 calc.out PID=8855 Stopped-->Running    CPU_Eff=92.62404   Throughput=4.19048
58 2016-12-7 19:25:6 ActiveProc=2 calc.out PID=8855 Running-->Exited    CPU_Eff=92.70402   Throughput=8.22741
59 2016-12-7 19:25:6 ActiveProc=1 calc1.out PID=8825 Stopped-->Running    CPU_Eff=92.60625   Throughput=8.21953
60 2016-12-7 19:25:7 ActiveProc=1 calc1.out PID=8825 Running-->Exited    CPU_Eff=92.76457   Throughput=11.98478
```

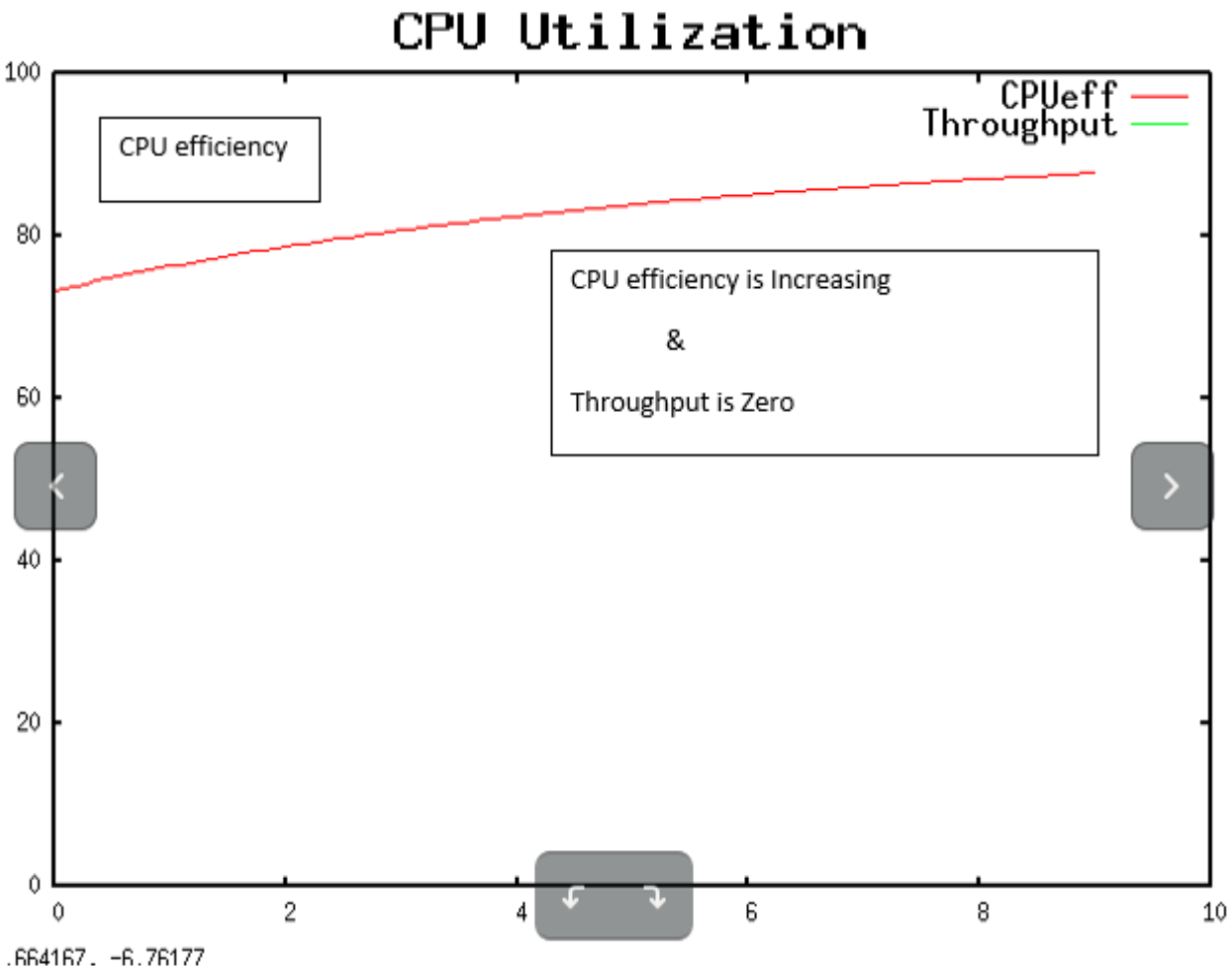
No. of Running Process is decreased when a process is terminated or completes its execution

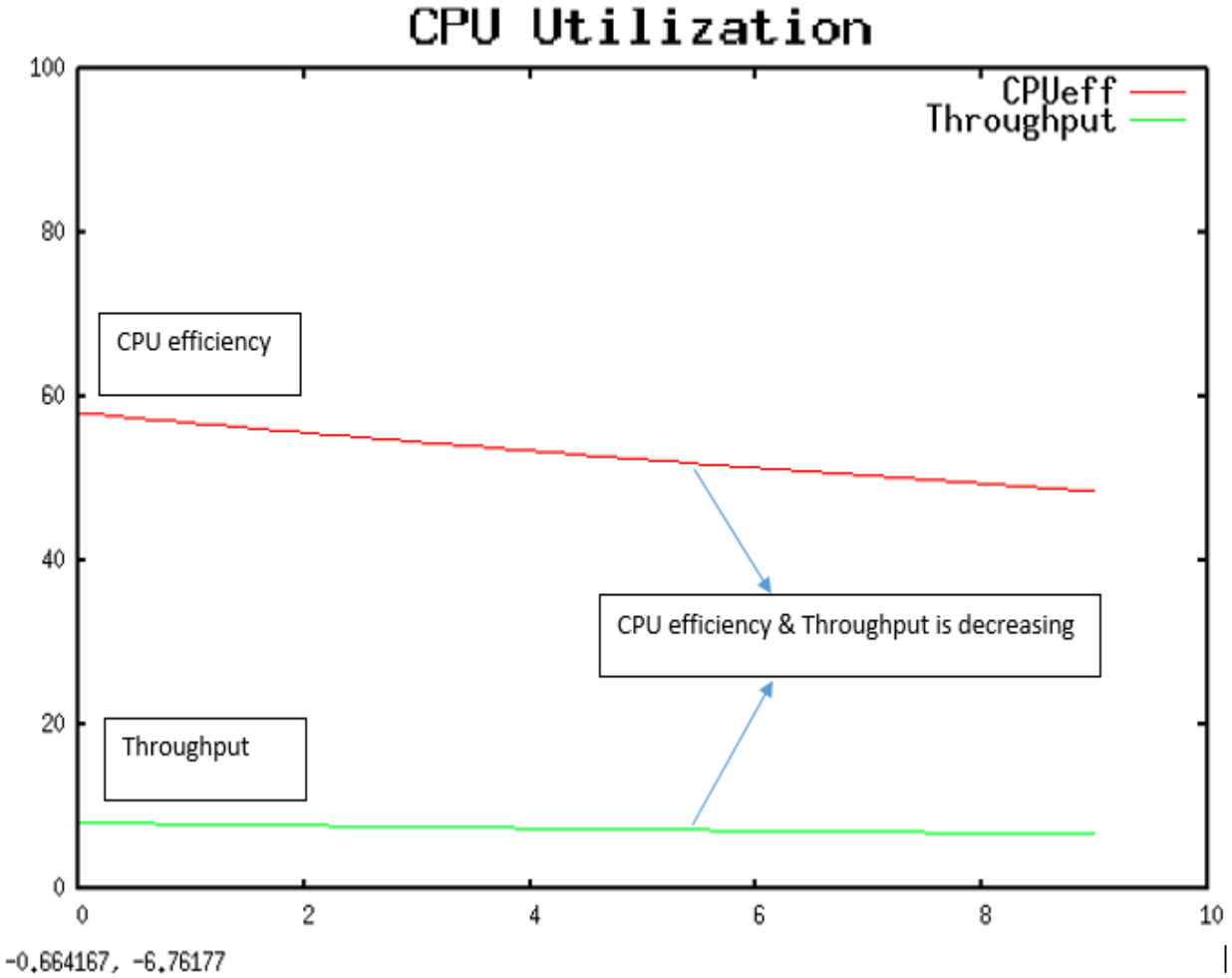



```
 1 2016-12-7 19:24:43 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=0.00000   Throughput=0.00000
 2 2016-12-7 19:24:44 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=0.00000   Throughput=0.00000
 3 2016-12-7 19:24:44 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=36.95515   Throughput=0.00000
 4 2016-12-7 19:24:45 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=27.00398   Throughput=0.00000
 5 2016-12-7 19:24:45 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=53.72089   Throughput=0.00000
 6 2016-12-7 19:24:46 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=42.31968   Throughput=0.00000
 7 2016-12-7 19:24:46 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=63.37492   Throughput=0.00000
 8 2016-12-7 19:24:47 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=52.33880   Throughput=0.00000
 9 2016-12-7 19:24:47 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=69.56268   Throughput=0.00000
10 2016-12-7 19:24:47 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=59.28686   Throughput=0.00000
11 2016-12-7 19:24:47 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=73.88167   Throughput=0.00000
12 2016-12-7 19:24:48 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=64.35210   Throughput=0.00000
13 2016-12-7 19:24:48 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=77.10324   Throughput=0.00000
14 2016-12-7 19:24:49 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=68.35179   Throughput=0.00000
15 2016-12-7 19:24:49 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=79.57847   Throughput=0.00000
16 2016-12-7 19:24:50 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=71.41828   Throughput=0.00000
17 2016-12-7 19:24:50 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=81.54992   Throughput=0.00000
18 2016-12-7 19:24:51 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=74.02406   Throughput=0.00000
19 2016-12-7 19:24:51 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=83.14182   Throughput=0.00000
20 2016-12-7 19:24:52 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=76.11245   Throughput=0.00000
21 2016-12-7 19:24:52 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=84.46237   Throughput=0.00000
22 2016-12-7 19:24:52 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=77.73281   Throughput=0.00000
23 2016-12-7 19:24:52 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=85.58917   Throughput=0.00000
24 2016-12-7 19:24:53 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=79.44978   Throughput=0.00000
25 2016-12-7 19:24:53 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=86.51639   Throughput=0.00000
26 2016-12-7 19:24:54 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=80.69899   Throughput=0.00000
27 2016-12-7 19:24:54 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=87.33173   Throughput=0.00000
28 2016-12-7 19:24:55 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=81.85949   Throughput=0.00000
29 2016-12-7 19:24:55 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=87.99328   Throughput=0.00000
30 2016-12-7 19:24:56 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=82.79027   Throughput=0.00000
31 2016-12-7 19:24:56 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=88.59695   Throughput=0.00000
32 2016-12-7 19:24:57 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=83.64474   Throughput=0.00000
33 2016-12-7 19:24:57 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=89.14794   Throughput=0.00000
34 2016-12-7 19:24:57 ActiveProc=3 calc.out PID=8855 Running-->Stopped    CPU_Eff=84.45797   Throughput=0.00000
35 2016-12-7 19:24:57 ActiveProc=3 calc1.out PID=8825 Stopped-->Running    CPU_Eff=89.65691   Throughput=0.00000
36 2016-12-7 19:24:58 ActiveProc=3 calc1.out PID=8825 Running-->Stopped    CPU_Eff=85.18222   Throughput=0.00000
37 2016-12-7 19:24:58 ActiveProc=3 calc2.out PID=8840 Stopped-->Running    CPU_Eff=90.10844   Throughput=0.00000
38 2016-12-7 19:24:59 ActiveProc=3 calc2.out PID=8840 Running-->Stopped    CPU_Eff=85.78964   Throughput=0.00000
39 2016-12-7 19:24:59 ActiveProc=3 calc.out PID=8855 Stopped-->Running    CPU_Eff=90.53065   Throughput=0.00000
```



Above image is the process image of individual 3 processes. Our process image contains the process ID, process state and its variable values. As we have taken a screen shot of process image at run time so as from above image you can see that 2 of them processes are in stopped state and 1 of them process is in running state.


CPU Utilization
CPUeff
Throughput
CPU efficiency
CPU efficiency is Increasing
&
Throughput is Zero
100
80
60
40
20
0
0
2
4
6
8
10
.664167, -6.76177


CPU Utilization
CPUeff
Throughput
CPU efficiency
CPU efficiency & Throughput is decreasing
Throughput
100
80
60
40
20
0
0
2
4
6
8
10
-0.664167, -6.76177

References

- http://unix.stackexchange.com/questions/16738/when-a-process-will-go-to-d-state
- http://unix.stackexchange.com/questions/2879/how-to-get-a-program-running-with-root-privileges-without-using-su-or-sudo
- http://unix.stackexchange.com/questions/5642/what-if-kill-9-does-not-work
- http://www.unix.com/programming/173333-how-sleep-wake-thread.html

Group members

Madhav Chavda – 1401006

Nisarg Tike – 1401070

Yash Kotadia – 1401114

Akshat Doshi – 1401119

THANK YOU