Lab Exercise - 6

 AIM :: WAP in shell script to implement CPU scheduling for `shortest job first` (sjf).

```
Source_Code ::
```

```
echo $'\n' "5C6 - Amit Singhal (11614802722)" $'\n'
read -p "Enter the number of processes: " num_processes
echo $'\n' "Enter Arrival Time & Burst Time for $num_processes processes"
# Collect process details
for ((i=0;i<num_processes;i++)); do</pre>
  echo -n "P$((i+1)): "
  read arrival_time burst_time
  processes[$i]="$arrival_time $burst_time"
done
# Initialize variables
total_completion_time=0
total_waiting_time=0
total_turnaround_time=0
completed_processes=0
gantt_chart="0" # Start Gantt chart at time 0
time=0
# Create an array to store completion status of each process (0 = incomplete, 1
= complete)
for ((i=0;i<num_processes;i++)); do</pre>
```

```
process_completed[$i]=0
done
# Function to find the process with the shortest burst time among those that
have arrived
find_shortest_job() {
  local min_burst=-1
  local min index=-1
  for ((i=0;i<num_processes;i++)); do
    current_process=(${processes[$i]})
    current_arrival_time=${current_process[0]}
    current_burst_time=${current_process[1]}
    if (( process_completed[$i] == 0 && current_arrival_time <= time ));</pre>
then
       if (( min_burst == -1 || current_burst_time < min_burst )); then
         min_burst=$current_burst_time
         min index=$i
       fi
    fi
  done
  echo $min_index
}
# Display table header
echo -e "\nProcess Arrival Time Burst Time Completion Time
Turnaround Time
                   Waiting Time"
# Process all processes using SJF
while (( completed_processes < num_processes )); do
  shortest_job=$(find_shortest_job)
  if (( shortest_job == -1 )); then
```

```
# No process available, increase time (idle)
    gantt_chart+=" -- XX -- $((++time))"
  else
    current_process=(${processes[$shortest_job]})
    current_arrival_time=${current_process[0]}
    current_burst_time=${current_process[1]}
    if (( time < current_arrival_time )); then</pre>
      time=$current_arrival_time
      gantt_chart+=" -- XX -- $time"
    fi
    completion_time=$((time + current_burst_time))
    turnaround_time=$((completion_time - current_arrival_time))
    waiting_time=$((turnaround_time - current_burst_time))
    # Update total values
    total_completion_time=$completion_time
    total_waiting_time=$((total_waiting_time + waiting_time))
    total_turnaround_time=$((total_turnaround_time + turnaround_time))
    # Mark the process as completed
    process_completed[$shortest_job]=1
    completed_processes = $((completed_processes + 1))
    # Display process details
    echo -e "P$((shortest_job+1))\t\t$current_arrival_time\t\
t$current_burst_time\t\t$completion_time\t\t $turnaround_time\t\t
$waiting_time"
    # Update Gantt chart
    gantt_chart+=" -- P$((shortest_job+1)) -- $completion_time"
    # Update current time
    time=$completion_time
```

done

```
# Calculate averages
avg_waiting_time=$(awk "BEGIN {printf \"%.2f\",
$total_waiting_time/$num_processes}")
avg_turnaround_time=$(awk "BEGIN {printf \"%.2f\",
$total_turnaround_time/$num_processes}")

# Display Gantt chart
echo -e "\nGantt Chart:"
echo -e "$gantt_chart"

# Display averages
echo ""
echo "Avg waiting time: $avg_waiting_time"
echo "Avg turnaround time: $avg_turnaround_time"
```

Output ::

```
singhal-amit@singhal-amit-ThinkPad-T430:~/Downloads/_LAB_Work/OS/Code$ vi prg_6_sjf.sh
singhal-amit@singhal-amit-ThinkPad-T430:~/Downloads/_LAB_Work/OS/Code$ chmod +x prg_6_sjf.sh
singhal-amit@singhal-amit-ThinkPad-T430:~/Downloads/_LAB_Wbrk/OS/Code$ ./prg 6 sjf.sh
 5C6 - Amit Singhal (11614802722)
Enter the number of processes: 4
 Enter Arrival Time & Burst Time for 4 processes
P1: 1 3
P2: 2 4
P3: 1 2
P4: 4 4
          Arrival Time
                        Burst Time
                                      Completion Time Turnaround Time
                                                                               Waiting Time
Process
P3
               1
                                2
                                                3
                                                                 2
                                                                                   0
                                3
                                                                 5
                                                                                   2
P1
                                                6
P2
                2
                                                10
                                                                 8
                                                                                   4
P4
                                4
                                                                                   6
                                                14
                                                                 10
Gantt Chart:
0 -- XX -- 1 -- P3 -- 3 -- P1 -- 6 -- P2 -- 10 -- P4 -- 14
Avg waiting time: 3.00
Avg turnaround time: 6.25
```