**Lab Exercise – 5**

* AIM ::

WAP in shell script to implement CPU scheduling for `first come first serve` (fcfs).

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

### echo -n "P$((i+1)): "

### read arrival\_time burst\_time

### processes[$i]="$arrival\_time $burst\_time"

### done

### # Sort processes by arrival time

### IFS=$'\n' sorted\_processes=($(sort -n -k1 <<<"${processes[\*]}"))

### unset IFS

### # Initialize variables

### total\_completion\_time=0

### total\_waiting\_time=0

### total\_turnaround\_time=0

### gantt\_chart="0" # Start Gantt chart at time 0

### # Display table header

### echo -e "\nProcess Arrival Time Burst Time Completion Time TurnAround Time Waiting Time"

### # Process all processes

### for ((i=0;i<num\_processes;i++)); do

### current\_process=(${sorted\_processes[$i]})

### current\_arrival\_time=${current\_process[0]}

### current\_burst\_time=${current\_process[1]}

### # If the process arrives after the last completion time, idle CPU

### if (( total\_completion\_time < current\_arrival\_time )); then

### idle\_time=$((current\_arrival\_time - total\_completion\_time))

### total\_completion\_time=$current\_arrival\_time

### gantt\_chart+=" -- XX -- $total\_completion\_time"

### fi

### # Calculate waiting time

### if (( total\_completion\_time >= current\_arrival\_time )); then

### waiting\_time=$((total\_completion\_time - current\_arrival\_time))

### else

### waiting\_time=0

### fi

### # Calculate completion time and turnaround time

### completion\_time=$((total\_completion\_time + current\_burst\_time))

### turnaround\_time=$((completion\_time - current\_arrival\_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))

### # Display process details

### echo -e "P$((i+1))\t\t$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$((i+1)) -- $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 ::