

```
In [2]: # Shortest Remaining Time First (SRTF)

# Function to find the waiting time
# for all processes
def findWaitingTime(processes, n, wt):
    rt = [0] * n

    # Copy the burst time into rt[]
    for i in range(n):
        rt[i] = processes[i][1]
    complete = 0
    t = 0
    minm = 999999999
    short = 0
    check = False

    # Process until all processes gets
    # completed
    while (complete != n):

        # Find process with minimum remaining
        # time among the processes that
        # arrives till the current time
        for j in range(n):
            if ((processes[j][2] <= t) and
                (rt[j] < minm) and rt[j] > 0):
                minm = rt[j]
                short = j
                check = True
        if (check == False):
            t += 1
            continue

        # Reduce remaining time by one
        rt[short] -= 1

        # Update minimum
        minm = rt[short]
        if (minm == 0):
            minm = 99999999

        # If a process gets completely
        # executed
        if (rt[short] == 0):

            # Increment complete
            complete += 1
            check = False

            # Find finish time of current
            # process
            fint = t + 1

            # Calculate waiting time
```

```

        wt[short] = (fint - proc[short][1] -
                      proc[short][2])

        if (wt[short] < 0):
            wt[short] = 0

        # Increment time
        t += 1

# Function to calculate turn around time
def findTurnAroundTime(processes, n, wt, tat):

    # Calculating turnaround time
    for i in range(n):
        tat[i] = processes[i][1] + wt[i]

# Function to calculate average waiting
# and turn-around times.
def findavgTime(processes, n):
    wt = [0] * n
    tat = [0] * n

    # Function to find waiting time
    # of all processes
    findWaitingTime(processes, n, wt)

    # Function to find turn around time
    # for all processes
    findTurnAroundTime(processes, n, wt, tat)

    # Display processes along with all details
    print("Processes Burst Time      Waiting",
          "Time      Turn-Around Time")
    total_wt = 0
    total_tat = 0
    for i in range(n):

        total_wt = total_wt + wt[i]
        total_tat = total_tat + tat[i]
        print(" ", processes[i][0], "\t\t",
              processes[i][1], "\t\t",
              wt[i], "\t\t", tat[i])

    print("\nAverage waiting time = %.5f "%(total_wt /n) )
    print("Average turn around time = ", total_tat / n)

# Driver code
if __name__ == "__main__":

    # Process id's
    proc = [[1, 6, 1], [2, 8, 1],
            [3, 7, 2], [4, 3, 3]]
    n = 4
    findavgTime(proc, n)

```

Processes	Burst Time	Waiting Time	Turn-Around Time
1	6	3	9
2	8	16	24
3	7	8	15
4	3	0	3

Average waiting time = 6.75000

Average turn around time = 12.75

In [ ]: