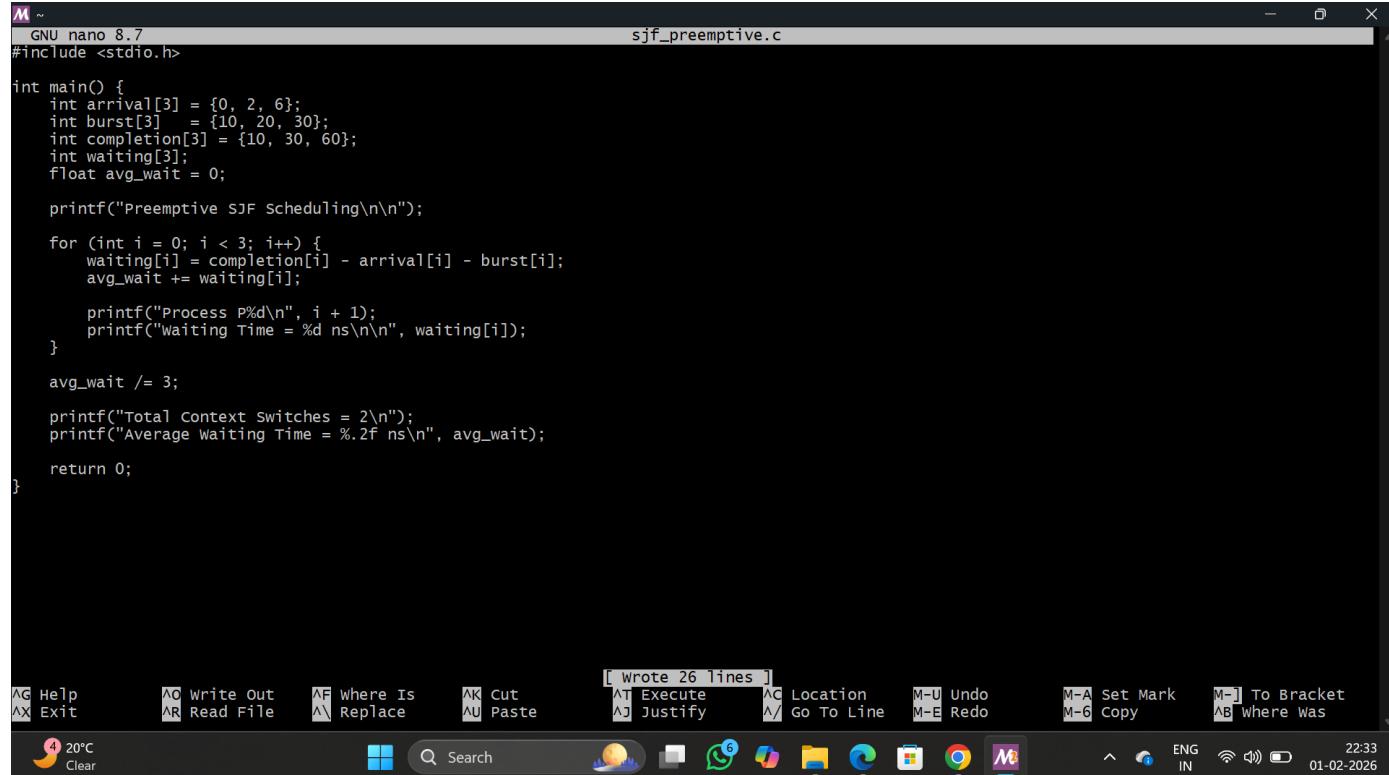


In an operating system three CPU-intensive processes are ready for execution, which require 10ns, 20ns and 30ns and arrival at times 0ns, 2ns and 6ns, respectively. Write a Program to calculate the total number of context switches needed if the operating system implements a shortest job first (preemptive) scheduling algorithm. Also calculate the average time for which the processes have to wait before getting the CPU.



```
M ~
GNU nano 8.7 sjf_preemptive.c
#include <stdio.h>

int main() {
    int arrival[3] = {0, 2, 6};
    int burst[3] = {10, 20, 30};
    int completion[3] = {10, 30, 60};
    int waiting[3];
    float avg_wait = 0;

    printf("Preemptive SJF scheduling\n\n");

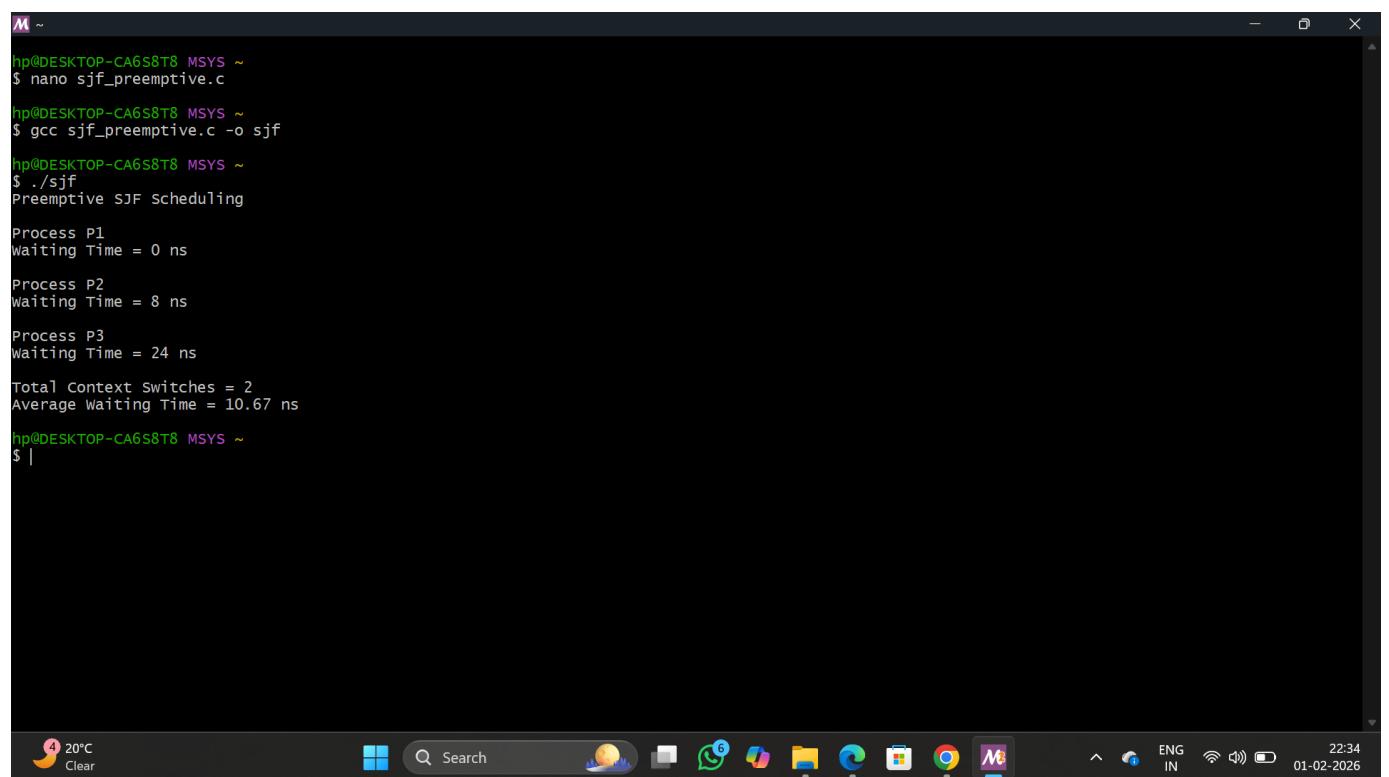
    for (int i = 0; i < 3; i++) {
        waiting[i] = completion[i] - arrival[i] - burst[i];
        avg_wait += waiting[i];

        printf("Process P%d\n", i + 1);
        printf("Waiting Time = %d ns\n", waiting[i]);
    }

    avg_wait /= 3;

    printf("Total Context Switches = 2\n");
    printf("Average Waiting Time = %.2f ns\n", avg_wait);

    return 0;
}
```



```
M ~
hp@DESKTOP-CA6S8T8 MSYS ~
$ nano sjf_preemptive.c
hp@DESKTOP-CA6S8T8 MSYS ~
$ gcc sjf_preemptive.c -o sjf
hp@DESKTOP-CA6S8T8 MSYS ~
$ ./sjf
Preemptive SJF Scheduling

Process P1
Waiting Time = 0 ns

Process P2
Waiting Time = 8 ns

Process P3
Waiting Time = 24 ns

Total Context Switches = 2
Average Waiting Time = 10.67 ns
hp@DESKTOP-CA6S8T8 MSYS ~
$ |
```