5 .Construct a scheduling program with C that selects the waiting process with the highest priority to execute next.

A. Code:

```
#include <stdio.h>

struct priority_scheduling {
   char process_name;
   int burst_time;
   int waiting_time;
   int turn_around_time;
   int priority;
};

int main() {
   int number_of_process;
   int total_waiting_time = 0, total_turnaround_time = 0;
   struct priority_scheduling temp_process;
   int ASCII_number = 65; // Start with 'A'
   int position;
   float average_waiting_time, average_turnaround_time;

   // Input the number of processes
   printf("Enter the total number of processes: ");
   scanf("%d", &number_of_process);

   struct priority_scheduling process[number_of_process];

   // Input burst time and priority for each process
   printf("\nPlease Enter the Burst Time and Priority of each process:\n");
   for (int i = 0; i < number_of_process; i++) {
      process[i].process_name = (char)ASCII_number;
      printf("\nEnter the details of process %c\n", process[i].process_name);
      printf("Enter the burst time: ");
      scanf("%d", &process[i].burst_time);
      printf("Enter the priority: ");
      scanf("%d", &process[i].priority);
      ASCII_number++;
   }

   // Sort processes based on priority (higher priority comes first)
   for (int i = 0; i < number_of_process; i++) {
      position = i;
      for (int j = i + 1; j < number_of_process; j++) {
         if (process[j].priority > process[position].priority) {
            position = j;
         }
      }
```

```
        temp_process = process[i];
        process[i] = process[position];
        process[position] = temp_process;
    }

    // Calculate waiting times
    process[0].waiting_time = 0; // First process has no waiting time
    for (int i = 1; i < number_of_process; i++) {
        process[i].waiting_time = process[i - 1].waiting_time + process[i - 1].burst_time;
        total_waiting_time += process[i].waiting_time;
    }

    // Calculate turnaround times
    for (int i = 0; i < number_of_process; i++) {
        process[i].turn_around_time = process[i].burst_time + process[i].waiting_time;
        total_turnaround_time += process[i].turn_around_time;
    }

    // Calculate averages
    average_waiting_time = (float)total_waiting_time / number_of_process;
    average_turnaround_time = (float)total_turnaround_time / number_of_process;

    // Display the results
    printf("\n\nProcess Name\tBurst Time\tWaiting Time\tTurnaround Time\n");
    printf("-------------------------------------------------------------\n");
    for (int i = 0; i < number_of_process; i++) {
        printf("\t%c\t\t%d\t\t%d\t\t%d\n",
            process[i].process_name, process[i].burst_time, process[i].waiting_time,
process[i].turn_around_time);
    }

    // Print averages
    printf("\n\nAverage Waiting Time: %.2f", average_waiting_time);
    printf("\nAverage Turnaround Time: %.2f\n", average_turnaround_time);

    return 0;
}
```

Output:

```
Please Enter the Burst Time and Priority of each process:

Enter the details of process A
Enter the burst time: 10
Enter the priority: 2

Enter the details of process B
Enter the burst time: 4
Enter the priority: 3

Enter the details of process C
Enter the burst time: 5
Enter the priority: 1

Enter the details of process D
Enter the burst time: 3
Enter the priority: 4


Process Name    Burst Time      Waiting Time    Turnaround Time
-------------------------------------------------------------
        D               3               0               3
        B               4               3               7
        A               10              7               17
        C               5               17              22


Average Waiting Time: 6.75
Average Turnaround Time: 12.25
```

```
Enter number of process:5
Enter Burst Time:
p1:12
p2:4
p3:5
p4:6
p5:8
Processt Burst Time tWaiting TimetTurnaround Time
p2tt 4tt 0ttt4
p3tt 5tt 4ttt9
p4tt 6tt 9ttt15
p5tt 8tt 15ttt23
p1tt 12tt 23ttt35
Average Waiting Time=10.200000Average Turnaround Time=17.200001n
--------------------------------
Process exited after 21.49 seconds with return value 0
Press any key to continue . . .
```