**Code(non-preemptive priority scheduling)**:

```
package com.company;


import java.util.Arrays;
import java.util.Comparator;
import java.util.Scanner;


 public class Main {
   static Scanner in = new Scanner(System.in);
   public static void main(String[] args) {
      int[] process = setBurstTime();
      int[] priority = setPriority(process.length);
      int[][] test = new int[process.length][2];
      for(int i=0;i<process.length;i++) {
         test[i][0] = priority[i];
         test[i][1] = process[i];
      }
      displayProcesses(process, priority);
      calcWaitingTime(test, 0);
   }
   static int[] setBurstTime(){
      System.out.print("Enter the no. of processes: ");
      int n = in.nextInt();
      int[] arr = new int[n];


      System.out.print("\nEnter burst time for each process: ");
      for (int i = 0;  i < arr.length; i++) {
         arr[i] = in.nextInt();
      }
      return arr;
   }
   static int[] setPriority(int n) {
      int[] priority = new int[n];
      for (int i = 0; i < priority.length; i++){
         System.out.print("Enter priority of P"+ (i + 1) + ": ");
         priority[i] = in.nextInt();
```

```
    }
    return priority;
}
static void displayProcesses(int[] arr1, int[] arr2){
    System.out.println("|  Process\t\t|\tBurst Time  |\tPriority  |");
    for (int i = 0; i < arr1.length ; i++) {
        System.out.println("|  P"+ (i + 1) + "\t\t\t|\t" + arr1[i] + "\t\t\t|\t" + arr2[i] + "\t\t  |");
    }
    System.out.println("+-------------------------------------------+");
}
static void calcWaitingTime(int[][] arr, int basedOnIndex){
    int waitingTime, sumWT, sumTAT;
    waitingTime = sumWT = sumTAT = 0;
    Arrays.sort(arr, new Comparator<int[]>() {
        @Override
        public int compare(int[] o1, int[] o2) {
            if(o1[basedOnIndex] > o2[basedOnIndex]) {
                return 1;
            }
            return -1;
        }
    });

    System.out.println("Waiting time & Turn around time:\n|   Process\t\t|  Waiting Time  |  Turn Around Time    |");
    for (int i = 0; i < arr.length; i++) {
        int turnAroundTime = 0;
        if(i != 0)
            waitingTime += arr[i-1][1];
        int j = i;
        while( j >= 0){
            turnAroundTime += arr[j][1];
            j--;
        }
        System.out.println("|  P"+ (i + 1) + "\t\t\t|  " + waitingTime + "\t\t\t |  " + turnAroundTime + "\t\t\t\t\t|");
        sumWT += waitingTime;
        sumTAT += turnAroundTime;
```

```
    }
    System.out.println("+-------------------------------------------------------+");
    System.out.println("Average waiting time = " + (float)sumWT / arr.length);
    System.out.println("Average turn around time = " + (float)sumTAT / arr.length);
  }
}
```

# OUTPUT:-

```
Enter the no. of processes: 3

Enter burst time for each process: 6 8 4
Enter priority of P1: 2
Enter priority of P2: 1
Enter priority of P3: 3
|  Process      |   Burst Time  |   Priority  |
|  P1           |   6           |   2         |
|  P2           |   8           |   1         |
|  P3           |   4           |   3         |
+-----------------------------------------------+
Waiting time & Turn around time:
|   Process     |  Waiting Time  |  Turn Around Time    |
|  P1           |  0             |  8                   |
|  P2           |  8             |  14                  |
|  P3           |  14            |  18                  |
+-------------------------------------------------------+
Average waiting time = 7.3333335
Average turn around time = 13.333333

Process finished with exit code 0
```