import java.util.Arrays;

import java.util.Scanner;

public class Priority {

public static void main(String[] args) {

System.out.println("\*\*\* Priority Scheduling \*\*\*");

// Ask the user to enter the number of processes

System.out.print("Enter Number of Process: ");

Scanner sc = new Scanner(System.in);

int numberOfProcess = sc.nextInt();

// Create arrays to store process names, burst times, and priorities

String process[] = new String[numberOfProcess]; // Process names (e.g., P1, P2, ...)

int burstTime[] = new int[numberOfProcess]; // Burst times of each process

int priority[] = new int[numberOfProcess]; // Priorities of each process

// Generate process names (P1, P2, P3, ..., Pn)

int p = 1;

for (int i = 0; i < numberOfProcess; i++) {

process[i] = "P" + p;

p++;

}

System.out.println(Arrays.toString(process));

// Ask the user to enter burst times for each process

System.out.print("Enter Burst Time for " + numberOfProcess + " process: ");

for (int i = 0; i < numberOfProcess; i++) {

burstTime[i] = sc.nextInt();

}

System.out.println(Arrays.toString(burstTime));

// Ask the user to enter priorities for each process

System.out.print("Enter Priority for " + numberOfProcess + " process: ");

for (int i = 0; i < numberOfProcess; i++) {

priority[i] = sc.nextInt();

}

System.out.println(Arrays.toString(priority));

// Sorting the processes, burst times, and priorities based on priority

int temp;

String temp2;

for (int i = 0; i < numberOfProcess - 1; i++) {

for (int j = 0; j < numberOfProcess - 1; j++) {

if (priority[j] > priority[j + 1]) {

// Swap priority values

temp = priority[j];

priority[j] = priority[j + 1];

priority[j + 1] = temp;

// Swap burst times

temp = burstTime[j];

burstTime[j] = burstTime[j + 1];

burstTime[j + 1] = temp;

// Swap process names

temp2 = process[j];

process[j] = process[j + 1];

process[j + 1] = temp2;

}

}

}

// Create arrays to store Turnaround Time (TAT) and Waiting Time

int TAT[] = new int[numberOfProcess + 1];

int waitingTime[] = new int[numberOfProcess + 1];

// Calculate Waiting Time & Turnaround Time

for (int i = 0; i < numberOfProcess; i++) {

TAT[i] = burstTime[i] + waitingTime[i];

waitingTime[i + 1] = TAT[i];

}

// Calculate the total Waiting Time and total Turnaround Time for all processes

int totalWT = 0;

int totalTAT = 0;

for (int i = 0; i < numberOfProcess; i++) {

totalTAT += (waitingTime[i] + burstTime[i]);

totalWT += waitingTime[i];

}

// Calculate the average Waiting Time and average Turnaround Time

double avgWT = totalWT / (double) numberOfProcess;

double avgTAT = totalTAT / (double) numberOfProcess;

// Display the process details in tabular form

System.out.println("Process BT WT TAT");

for (int i = 0; i < numberOfProcess; i++) {

System.out.println(process[i] + " " + burstTime[i] + " " + waitingTime[i] + " " + TAT[i]);

}

// Display the average Waiting Time and average Turnaround Time

System.out.println("\nAverage Waiting Time: " + avgWT);

System.out.println("Average Turnaround Time: " + avgTAT);

}

}