#include <stdio.h>

#include <stdlib.h>

#define MAX 100

void priorityPreemptive(int n, int at[], int bt[], int pr[]) {

    int ct[n], tat[n], wt[n], rem\_bt[n], is\_completed[n];

    int time = 0, completed = 0, min\_priority, index;

    for (int i = 0; i < n; i++) {

        rem\_bt[i] = bt[i];

        is\_completed[i] = 0;

    }

    while (completed < n) {

        min\_priority = 9999;

        index = -1;

        for (int i = 0; i < n; i++) {

            if (at[i] <= time && is\_completed[i] == 0 && pr[i] < min\_priority && rem\_bt[i] > 0) {

                min\_priority = pr[i];

                index = i;

            }

        }

        if (index == -1) {

            time++;

        } else {

            rem\_bt[index]--;

            time++;

            if (rem\_bt[index] == 0) {

                ct[index] = time;

                is\_completed[index] = 1;

                completed++;

            }

        }

    }

    float total\_tat = 0, total\_wt = 0;

    printf("\nP#\tAT\tBT\tPR\tCT\tTAT\tWT\n");

    for (int i = 0; i < n; i++) {

        tat[i] = ct[i] - at[i];

        wt[i] = tat[i] - bt[i];

        total\_tat += tat[i];

        total\_wt += wt[i];

        printf("%d\t%d\t%d\t%d\t%d\t%d\t%d\n", i + 1, at[i], bt[i], pr[i], ct[i], tat[i], wt[i]);

    }

    printf("Average TAT: %.2f\n", total\_tat / n);

    printf("Average WT: %.2f\n", total\_wt / n);

}

void priorityNonPreemptive(int n, int at[], int bt[], int pr[]) {

    int ct[n], tat[n], wt[n], is\_completed[n], rem\_bt[n];

    int time = 0, completed = 0;

    for (int i = 0; i < n; i++) {

        is\_completed[i] = 0;

        rem\_bt[i] = bt[i];

    }

    while (completed < n) {

        int min\_priority = 9999, index = -1;

        for (int i = 0; i < n; i++) {

            if (at[i] <= time && is\_completed[i] == 0 && pr[i] < min\_priority) {

                min\_priority = pr[i];

                index = i;

            }

        }

        if (index == -1) {

            time++;

        } else {

            time += bt[index];

            ct[index] = time;

            is\_completed[index] = 1;

            completed++;

        }

    }

    float total\_tat = 0, total\_wt = 0;

    printf("\nP#\tAT\tBT\tPR\tCT\tTAT\tWT\n");

    for (int i = 0; i < n; i++) {

        tat[i] = ct[i] - at[i];

        wt[i] = tat[i] - bt[i];

        total\_tat += tat[i];

        total\_wt += wt[i];

        printf("%d\t%d\t%d\t%d\t%d\t%d\t%d\n", i + 1, at[i], bt[i], pr[i], ct[i], tat[i], wt[i]);

    }

    printf("Average TAT: %.2f\n", total\_tat / n);

    printf("Average WT: %.2f\n", total\_wt / n);

}

void sortProcesses(int n, int at[], int bt[], int pr[]) {

    int temp\_at, temp\_bt, temp\_pr;

    for (int i = 0; i < n - 1; i++) {

        for (int j = i + 1; j < n; j++) {

            if (at[i] > at[j] || (at[i] == at[j] && pr[i] > pr[j])) {

                temp\_at = at[i];

                at[i] = at[j];

                at[j] = temp\_at;

                temp\_bt = bt[i];

                bt[i] = bt[j];

                bt[j] = temp\_bt;

                temp\_pr = pr[i];

                pr[i] = pr[j];

                pr[j] = temp\_pr;

            }

        }

    }

}

int main() {

    int n, choice;

    printf("Enter number of processes: ");

    scanf("%d", &n);

    int at[n], bt[n], pr[n];

    for (int i = 0; i < n; i++) {

        printf("Enter AT, BT, and Priority P%d: ", i + 1);

        scanf("%d %d %d", &at[i], &bt[i], &pr[i]);

    }

    sortProcesses(n, at, bt, pr);

    printf("\nChoose Scheduling Algorithm:\n");

    printf("1. Preemptive Priority Scheduling\n");

    printf("2. Non-Preemptive Priority Scheduling\n");

    printf("Enter choice: ");

    scanf("%d", &choice);

    if (choice == 1) {

        priorityPreemptive(n, at, bt, pr);

    } else if (choice == 2) {

        priorityNonPreemptive(n, at, bt, pr);

    } else {

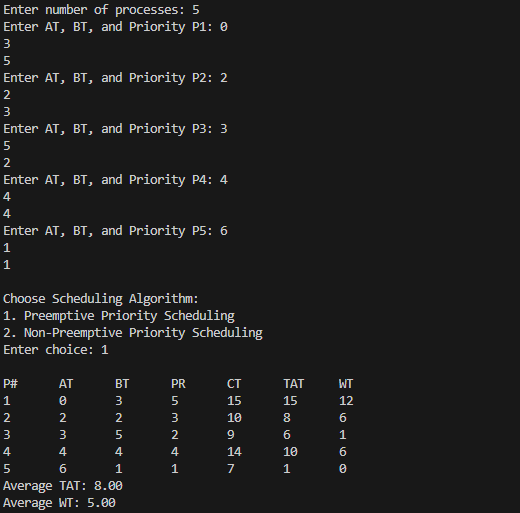
        printf("Invalid choice!\n");

    }

    return 0;

}

Pre emptive



Non preemptive

