Prints de execução

Arquivo de entrada

```
3
5 0
8 0
13 0
```

Código

```c
/*
MEMBROS:
André Matteucci - 32273541
Enzo Koji       - 32273754
Felipe Ribeiro  - 32212720
*/

#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>

typedef struct {
    int t;
    int d;
} rider;

void readData(char* filePath, rider* riders, int* n) {
    FILE* file = fopen(filePath, "r");
    if (file == NULL) {
        printf("Erro ao abrir o arquivo %s.\n", filePath);
        exit(1);
    }

    fscanf(file, "%d", n);
    for (int i = 0; i < *n; i++) {
        fscanf(file, "%d %d", &riders[i].t, &riders[i].d);
    }
    fclose(file);
}

int escalator(rider* riders, int n) {
    rider currentRider = riders[0];

    int mainIndex = 0;
    int auxIndex = 0;
```

```
    int estArrival;

    rider pendingRiders[10000];
    int remainingRiders = n;

    int lastMoment = 0;
    int direction = -1;
    int moment = 0;

    bool pending = false;

    while (remainingRiders > 0) {
        if (pending && (riders[mainIndex].t > estArrival || mainIndex >=
n)) {
            currentRider = pendingRiders[0];
            moment += 10;
            direction = currentRider.d;
            estArrival = moment + 10;
            remainingRiders--;
            pending = false;
        } else {
            currentRider = riders[mainIndex];

            if (direction == -1) {
                moment = currentRider.t < moment ? moment :
currentRider.t;
                direction = currentRider.d;
                estArrival = currentRider.t + 10;

                mainIndex++;
                remainingRiders--;
            } else if (direction == currentRider.d) {
                moment = currentRider.t;
                estArrival = currentRider.t + 10;

                mainIndex++;
                remainingRiders--;
            } else {
                if (riders[mainIndex + 1].t - riders[mainIndex].t >
riders[mainIndex - 1].t) {
                    moment = estArrival;
                    direction = -1;
                }
                else if (riders[mainIndex + 1].t <= estArrival) {
                    pendingRiders[0] = riders[mainIndex];
                    pending = true;
                    mainIndex++;
                }
```

```
            }
        }
    }

    moment += 10;
    lastMoment = moment;

    return lastMoment;
}

int main() {
    rider riders[10000];
    int n;

    readData("./entrada.txt", riders, &n);
    int lastMoment = escalator(riders, n);

    printf("O último momento em que a escada para é %d\n", lastMoment);
    return 0;
}
```

Execução

```
Run                                    59ms on 19:13:48, 11/21 ✓
O último momento em que a escada para é 23
```