

Tarea 2 - CIT1000 sección 12

Profesor: Marcello Tavano
Ayudantes: Ruben Hermosilla; Javier Molina

30 de Abril de 2021

Lea atentamente el siguiente enunciado y realice un programa que solucione la problemática. Recuerde leer bien y reflexionar sobre lo pedido antes de lanzarse al código. EL error más frecuente cometido por los alumnos es “desarrollar el problema equivocado”, esto es elaborar algo que no le están pidiendo.

Programa 1

Juan es un estadístico al cual le encanta maximizar su energía para realizar las tareas de su día a día, por lo que todas las mañanas se dedica a organizarlas. La manera en que realiza esto es la siguiente: Juan propone una cantidad de energía para su día y registra cada una de sus tareas con un nombre, un coste de energía y una prioridad, para finalmente ordenarlas por este último factor (prioridad).

Como Juan no sabe programación, le pidió a usted que realice un programa que lo ayude con la tarea de organizar su día. Dicho programa debe recibir una cantidad de energía E , una cantidad de tareas T y finalmente el nombre n , el coste de energía c y la prioridad p de cada una de las T tareas.

Finalmente el programa debe entregarle a Juan su lista de tareas ordenadas por prioridad (mayor a menor), el promedio de energía de las tareas del día y finalmente la energía que a Juan le sobraría (si es que sobra).

Hints:

- Recuerde validar la entrada, Juan no podrá realizar todas las tareas si no tiene suficiente energía.
- Para este ejercicio los arreglos son fundamentales, utilice los índices de éstos para facilitar la relación entre las tareas y sus características.
- Antes de lanzarte al código, visualiza!

0.1. Inputs

- **E** : Total de energía que Juan dispone para el día, donde $\mathbf{E} > 0$
- **T** : Cantidad de tareas que Juan quiere realizar en su día, donde $\mathbf{T} > 0$
- n_t, c_t, p_c : Nombre, costo y prioridad de cada tarea $t \in T$.
 - n_t es un string.
 - $c_t > 0$.
 - $1 \leq p_t \leq 100$.

0.2. Outputs

- Las **T** tareas representadas con sus 3 datos n_t, c_t, p_c ordenadas de mayor a menor por prioridad.
- **PE** : Energía promedio de las tareas del día.
- **ES** : Energía sobrante del día ($ES \geq 0$).

0.3. Ejemplos

Considere los siguientes ejemplos.

Ejemplo 1:

```
Ingrese energia de la persona.
0
Ingrese una cantidad mayor a 0.
1000
Ingrese cantidad de tareas a agregar
0
Ingrese cantidad de tareas mayor que 0.
3
Ingrese nombre, coste de energia y prioridad de la tarea 1.
Tarea1 0 0
Ingrese nombre, coste de energia y prioridad de la tarea 1 nuevamente.
Tarea1 0 2
Ingrese nombre, coste de energia y prioridad de la tarea 1 nuevamente.
Tarea1 200 2
Ingrese nombre, coste de energia y prioridad de la tarea 2.
Tarea2 700 6
Ingrese nombre, coste de energia y prioridad de la tarea 3.
Tarea3 80 7
El orden de las tareas es:
N ----> Nombre , Energia, Prioridad
1 ----> Tarea3 , 80 , 7
2 ----> Tarea2 , 700 , 6
3 ----> Tarea1 , 200 , 2
La energia sobrante es: 20
El promedio de energia que se gastara en el dia es: 326.667

Process returned 0 (0x0)   execution time : 57.552 s
Press any key to continue.
```

Figura 1: Ejemplo1

Como se puede ver, el programa debe pedir la energia inicial y limitar su entrada, después se ingresa la cantidad de tareas que se realizarán (también se limita la entrada) y finalmente, se ingresa el nombre , la energia y la prioridad de las T tareas, incluyendo las restricciones de energia y de prioridad.

Finalmente se hace entrega de los outputs pedidos.

Entrega

El formato de entrega será por medio de CANVAS, y el formato del archivo a entregar debe ser un .cpp. El nombre del archivo debe ser su rut por ejemplo (**23254324-3.rar**).

La fecha limite de entrega es el día (**Viernes 7 de Mayo a las 23:59 pm**).