

Fonaments de Programació

PAC4 - 20232

[CAT] Enunciat

Es demana que implementeu un programa en llenguatge C que **compari dues excursions** per la muntanya i indiqui quina d'elles és **la més exigent**.

Per a gestionar les dades d'una excursió usarem una **tupla** que contindrà el seu **nom**, la **distància recorreguda** i el **desnivell positiu acumulat**.

Una excursió serà més exigent que una altra si la seva distància recorreguda és superior. En cas d'empat, la més exigent serà la que tingui un major desnivell positiu acumulat. En cas d'empat, la més exigent serà la segona que s'hagi introduït.

Entrada de dades

El programa llegirà des de teclat les dades de la primera excursió i a continuació les dades de la segona. Per a cada excursió es demanarà:

- El nom, sense espais en blanc i amb una longitud màxima de 15 caràcters.
- La distància en quilòmetres, que pot contenir decimals.
- El desnivell positiu acumulat en metres, sense decimals.

Sortida de dades

El programa retornarà el **nom** de l'excursió **més exigent**.

Exemple d'execució

A l'annex del final d'aquest document podeu consultar un exemple d'execució del programa en llenguatge C: reviseu-lo per tal d'utilitzar els literals exactes per resoldre l'exercici.

Fundamentos de programación

PEC4 - 20232

[ES] Enunciado

Se pide que implementéis un programa en lenguaje C que **compare dos excursiones** por la montaña e indique cuál de ellas es **la más exigente**.

Para gestionar los datos de una excursión usaremos una **tupla** que contendrá su **nombre**, la **distancia recorrida** y el **desnivel positivo acumulado**.

Una excursión será más exigente que otra si su distancia recorrida es superior. En caso de empate, la más exigente será la que tenga un mayor desnivel positivo acumulado. En caso de empate, la más exigente será la segunda que se haya introducido.

Entrada de datos

El programa leerá desde teclado los datos de la primera excursión y a continuación los datos de la segunda. Para cada excursión se pedirá:

- El nombre, sin espacios en blanco y con una longitud máxima de 15 caracteres.
- La distancia en kilómetros, que puede contener decimales.
- El desnivel positivo acumulado en metros, sin decimales.

Salida de datos

El programa devolverá el **nombre** de la excursión **más exigente**.

Ejemplo de ejecución

En el anexo del final de este documento podéis consultar un ejemplo de ejecución del programa en lenguaje C: revisadlo para utilizar los literales exactos para resolver el ejercicio.

Programming fundamentals

CAA4 - 20232

[EN] Summary

You are asked to implement a program in C-language that **compares two hiking routes** and indicates which of them is **the hardest**.

To manage the data of a route we will use a **record** that will contain its **name**, the **distance traveled** and the **cumulative elevation difference**.

A route is harder than another if its distance traveled is longer. In case of a tie, the hardest will be the one with the greatest cumulative elevation difference. In case of a tie, the hardest will be the second one entered.

Data input

The program will read from the keyboard the data of the first route and then the data of the second one. For each route the program should request:

- The name, without spaces and with a maximum length of 15.
- The distance in kilometers, that may contain decimals.
- The cumulative elevation difference in meters, without decimals.

Data output

The program should return the **name** of the **hardest route**.

Execution example

In the annex at the end of this document you can see an example of C-program execution: review it in order to use the exact literals to solve the exercise.

[CAT] Annex

[ES] Anexo

[EN] Annex

INPUT

ROUTE 1

NAME?

OllaDeNuria

DISTANCE?

26.98

CUMULATIVE ELEVATION DIFFERENCE?

2027

ROUTE 2

NAME?

CanalCristall

DISTANCE?

16.84

CUMULATIVE ELEVATION DIFFERENCE?

1473

OUTPUT

THE HARDEST ROUTE IS OllaDeNuria