

Fundamentos de la programación I – grupos ***

Tiempo disponible: 3h

Una vuelta ciclista suele durar varias semanas, durante las cuales cada día se corre una etapa. Dada la clasificación de una vuelta, queremos actualizarla con los tiempos que han empleado los ciclistas en recorrer una de las etapas y obtener el ganador de dicha etapa. También consultaremos los tiempos obtenidos por los mejores ciclistas de alguno de los equipos.

Cada etapa la gana aquel ciclista que emplea menos tiempo en recorrerla. El ganador de la vuelta completa es aquel ciclista que emplea menos tiempo en recorrer todas las etapas (la suma de los tiempos que ha empleado en todas las etapas es menor que la suma de los tiempos de todos los otros ciclistas). En cada etapa algún ciclista puede abandonar, bien porque se lesiona o bien porque llega fuera de tiempo a la meta, es decir emplea demasiado tiempo en recorrer la etapa. Cuando un ciclista abandona una etapa queda fuera de la carrera y ya no corre más etapas. Cada ciclista pertenece a un equipo, de forma que durante la carrera los ciclistas del mismo equipo se ayudan. Hay también un premio para el mejor equipo de la carrera.

Cada ciclista está identificado por el dorsal con el que corre (número entre 1 y el número total de ciclistas inscritos en la carrera) y forma parte de un equipo. No todos los equipos tienen que tener el mismo número de ciclistas.

La información sobre los ciclistas se encuentra en un fichero de texto: `ciclistas.txt`, donde figuran todos los ciclistas inscritos en la carrera. Cada línea del fichero tiene la información de un ciclista. Primero el equipo al que pertenece el ciclista (cadena de caracteres sin blancos), su nombre (cadena de caracteres sin blancos), su número de dorsal (entero positivo), si todavía está en la carrera ("IN" o "OUT"), el número de etapas que ya ha corrido y, por último, el tiempo empleado en cada una de las etapas.

COFIDIS Herrada	4	IN	2	250	250
COFIDIS Fernandez	11	OUT	1	300	
INEOS Bernal	5	IN	2	300	250
INEOS Carapaz	6	IN	2	280	220
INEOS Yates	12	IN	2	300	200
JUMBO Dumoulin	3	IN	2	100	400
JUMBO Roglic	2	IN	2	100	100
JUMBO Laporte	8	OUT	0		
MOVISTAR Barta	9	IN	2	300	300
MOVISTAR Izagirre	10	IN	2	100	200
MOVISTAR Mas	1	IN	2	200	250
MOVISTAR Valverde	7	IN	2	120	130



La información sobre la última etapa que se ha corrido se encuentra en el fichero: `etapaN.txt`. El fichero tiene una línea por cada ciclista que ha salido en esta etapa, indicando el dorsal del ciclista y el tiempo que ha tardado en completar la etapa. Si el ciclista no terminó la etapa, o la terminó fuera de tiempo, figurará con el tiempo máximo para completar las etapas más uno (`MAX_TIEMPO+1`). El fichero termina con una línea con dos valores -1.

1	200
2	250
3	230
4	150
5	501
6	250
7	100
9	120
10	400
12	501
-1	-1

Detalles de la implementación.

- Constantes y tipos de datos (1 punto)
 - a) Se declaran constantes para el número de ciclistas que se inscriben en la vuelta: `NUM_CICLISTAS = 12`; el número de etapas que tiene la vuelta: `NUM_ETAPAS = 20`; y el tiempo máximo en que deben completar las etapas: `MAX_TIEMPO = 500`. Suponemos el mismo tiempo máximo para todas las etapas.
 - b) El tiempo empleado por cada ciclista en recorrer la última etapa se guarda en una lista, `tListaEtapa`, de tamaño máximo el número de ciclistas inscritos en la vuelta. La lista guarda información del dorsal de los ciclistas que han tomado la salida en esa etapa y el tiempo que han tardado en completarla.
 - c) La información sobre los ciclistas, `tCiclista`, incluye el equipo y el nombre del ciclista, su número de dorsal, el tiempo acumulado de todas las etapas que ha corrido, si el ciclista continúa en carrera, el número de etapas de la carrera que ha completado y un array de tamaño máximo `NUM_ETAPAS`, en el que se guarda el tiempo en que ha completado cada etapa. La información de todos los ciclistas se guarda en un array `tCiclistas` de tamaño `NUM_CICLISTAS`. Este array está ordenado por el nombre de los equipos.
- Se implementarán al menos las siguientes funciones:
 - 1) `cargaEtapa()`. Guarda la información del fichero `etapaN.txt` en la lista de tipo `tListaEtapa`. La información se guardará en el orden en que aparece en el fichero. Dará error si no se puede abrir el fichero. (1 punto).
 - 2) `ganadorEtapa()`. Devuelve el dorsal y el tiempo del ganador de la etapa, es decir el ciclista que ha empleado menos tiempo en recorrerla. (1 punto)
 - 3) `cargaCiclistas()`. Guarda la información del fichero `ciclistas.txt` en el array de tipo `tCiclistas`. Para cada ciclista debe calcular el tiempo acumulado que lleva en las etapas ya disputadas. La información se guardará en el orden en que aparece en el fichero. Dará error si no se puede abrir el fichero. (1 punto).
 - 4) `mostrarCiclistas()`. Muestra la información del array de ciclistas según el formato que se muestra en el ejemplo. Se utilizará una función `mostrarCiclista()` que muestre el contenido de cada ciclista. Observa que no se muestra tiempo total para los ciclistas que han abandonado. Se ha utilizado un ancho de campo de 10 para los 5 primeros valores. (1 punto)

- 5) `buscaCiclista()`. Busca en el array de ciclistas un dorsal que se proporciona como parámetro y devuelve su posición en el array. Si el dorsal no existe en el array se devuelve el valor -1. (1 punto).
- 6) `actualizaCiclistas()`. Actualiza el array de ciclistas con los tiempos de la última etapa que se ha corrido. Debe añadir el tiempo de la nueva etapa y modificar la suma de tiempos de todas las etapas. Si un ciclista no completó la etapa, o la completó fuera de tiempo, se marcará que el ciclista abandona y no se modificarán los tiempos que había obtenido hasta este momento. (1 punto)
- 7) `tiempoEquipo()`. Dado el array de ciclistas y el nombre de un equipo obtiene la suma del tiempo que han tardado los dos mejores ciclistas de ese equipo que no hayan abandonado. Se garantiza que todos los equipos al inscribirse tienen al menos dos ciclistas. Si el equipo no tiene dos ciclistas en carrera devuelve -1 (1,5 puntos)
- 8) `guardaCiclistas()`. Guarda la información del array de ciclistas en el fichero `ciclistas.txt` (0,5 puntos)

El programa principal carga los datos de la etapa y calcula el ganador de la etapa. A continuación carga los datos de los ciclistas, muestra los datos que ha cargado, actualiza la información con los datos de la etapa y vuelve a mostrar los datos de los ciclistas ya actualizados. Por último calcula el tiempo de los dos mejores ciclistas del equipo Movistar y del equipo Ineos y los muestra por consola. (1 punto)

Ejemplo:

El ganador de la etapa es el dorsal 7 con un tiempo de 100

Clasificación de la carrera antes de la última etapa

EQUIPO	NOMBRE	DORSAL	IN/OUT	TOTAL	TIEMPOS
COFIDIS	Herrada	4	IN	500	250 250
COFIDIS	Fernandez	11	OUT		300
INEOS	Bernal	5	IN	550	300 250
INEOS	Carapaz	6	IN	500	280 220
INEOS	Yates	12	IN	500	300 200
JUMBO	Dumoulin	3	IN	500	100 400
JUMBO	Roglic	2	IN	200	100 100
JUMBO	Laporte	8	OUT		
MOVISTAR	Barta	9	IN	600	300 300
MOVISTAR	Izagirre	10	IN	300	100 200
MOVISTAR	Mas	1	IN	450	200 250
MOVISTAR	Valverde	7	IN	250	120 130

Clasificación de la carrera después de la última etapa

EQUIPO	NOMBRE	DORSAL	IN/OUT	TOTAL	TIEMPOS
COFIDIS	Herrada	4	IN	650	250 250 150
COFIDIS	Fernandez	11	OUT		300
INEOS	Bernal	5	OUT		300 250
INEOS	Carapaz	6	IN	750	280 220 250
INEOS	Yates	12	OUT		300 200
JUMBO	Dumoulin	3	IN	730	100 400 230
JUMBO	Roglic	2	IN	450	100 100 250

JUMBO	Laporte	8	OUT				
MOVISTAR	Barta	9	IN	720	300	300	120
MOVISTAR	Izagirre	10	IN	700	100	200	400
MOVISTAR	Mas	1	IN	650	200	250	200
MOVISTAR	Valverde	7	IN	350	120	130	100

El tiempo de los dos mejores corredores del MOVISTAR es 1000

El equipo INEOS no tiene dos corredores en carrera

Recuerda que se valorará la eficiencia del código, la ausencia de código repetido, la correcta utilización de los esquemas vistos en clase, etc. y que no se permite el uso de variables globales, ni de instrucciones de salto salvo los return adecuados en las funciones.

Entrega del examen: El programa se enviará a través de una tarea del campus virtual.

