

PRÁCTICA 1 – El viaje de Frodo

Reunidos en Rivendel (hogar de Elrond) los representantes de las fuerzas del bien (hombres, elfos, enanos y hobbits), se decide que sea Frodo, un pequeño hobbit de la Comarca, el encargado de transportar el Anillo Único hasta el Monte del Destino, en las oscuras tierras de Mordor, allí donde fue forjado y donde deberá ser destruido.

El viaje será duro y extremadamente peligroso, ya que Sauron, señor de la oscuridad, controla Mordor y tiene su ejército de orcos y otras criaturas detestables repartidas por sus dominios.

Se trata de escribir un programa que represente el viaje que realiza nuestro amigo Frodo desde Rivendel hasta Mordor (si es que llega). Para ello, representaremos el terreno a recorrer mediante una cuadrícula de tamaño $N \times M$. Rivendel (el inicio) se encuentra en la posición $[1,1]$ de la cuadrícula, mientras que el Monte del Destino (el objetivo) se encuentra en la posición $[N,M]$.

Frodo solamente podrá avanzar siguiendo cuatro movimientos, que son norte, sur, este y oeste. No podrá pasar de una casilla a otra de la cuadrícula mediante un movimiento en diagonal.

Nuestro hobbit inicia su marcha con una determinada energía, expresada numéricamente, la cual irá consumiendo o recuperando según vaya recorriendo el camino.

En cada casilla de la cuadrícula Frodo se encontrará la dirección que tiene que tomar para realizar el siguiente movimiento. Además, en cada casilla se podrá encontrar las siguientes cosas:

- **Pan élfico** -> Se come el pan y recupera dos unidades de energía. Continúa su camino según la dirección que figure en la casilla.
- **Frutos de las zarzas de Mordor** -> Se come los frutos y entonces se mareo. Estas casillas ni dan ni consumen energía. Una vez que Frodo se ha mareado, los cinco movimientos siguientes que realice los hará en dirección contraria a lo que indique la casilla. Esto se aplica desde la casilla que contiene la zarza. Por ej, llega a la casilla $[i,j]$ que tiene dirección este. Esta casilla tiene zarza, por lo que irá al oeste. Cuando pase a la casilla $[i-1,j]$, irá en la dirección contraria que marque, y así hasta que realice otros 4 movimientos (5 movimientos en total).
El mareo es acumulativo, es decir, que si Frodo se mareo, recorre tres casillas (todavía le quedan 2 para que se le quite el mareo) y vuelve a caer en otra casilla con zarzas, entonces el mareo le aumentará a 7 casillas (las 2 que quedaban más las 5 del nuevo mareo).
- **Un orco** -> Pueden darse dos casos:
 - Si Frodo está lúcido, entonces lucha con el orco y acaba con él, por lo que puede seguir avanzando, pero consumirá tres unidades de energía debido a la lucha con el orco. Continúa su camino según la dirección que figure en la casilla.
 - Si Frodo está mareado, entonces perderá la vida a manos del orco, que le proporcionará una muerte lenta, sangrienta y dolorosa a nuestro pobre hobbit.
- **Nada** -> Si al moverse a una casilla, dicha casilla está vacía, pueden darse dos casos:
 - Si Frodo está lúcido, entonces consume una unidad de energía y sigue avanzando según la dirección que figure en la casilla.
 - Si Frodo está mareado, entonces consume dos unidades de energía y sigue avanzando en sentido contrario a la dirección que figura en la casilla.

Cada vez que se visita una casilla que no está vacía, después de visitarse queda vacía. Es decir, en las casillas que había pan élfico, frutos de zarzas (alimentos que Frodo se come) u orcos (si Frodo no muere es porque mata al orco gracias a su daga Dardo), una vez visitadas quedarán vacías, y su funcionamiento se convierte en el de una casilla en la que no hay nada, cuyo comportamiento ya se ha descrito.

El viaje puede terminar por tres motivos:

- Frodo alcanza la salida (casilla $[N,M]$) y puede destruir el anillo, venciendo a Sauron y derrotando el mal.
- Frodo se queda sin fuerzas. En el momento en que el contador de energía de Frodo llega a cero, independientemente de la casilla a la que pase, muere por agotamiento. Incluso, aunque llegue a la casilla $[N,M]$, si al realizar ese último movimiento el contador de energía se queda a cero, se considerará que Frodo ha muerto.
- Frodo muere a manos de un orco.

Funcionamiento de la práctica

Al iniciarse el programa, este pedirá, por el orden que aquí se indica, separados por “intro” y sin ningún paso intermedio¹, los siguientes datos:

- Número de filas de la matriz que representa el recorrido (N).
- Número de columnas de la matriz que representa el recorrido (M).
- Energía inicial con la que parte Frodo. Será un número entero entre 1 y $N \times M$.
- Direcciones y contenidos de las casillas de la matriz siguiendo el orden $[1,1],[1,2],\dots,[1,M],[2,1],[2,2],\dots,[2,M],\dots,[N,1],[N,2],\dots,[N,M]$. Las direcciones y contenido se representarán así:
 - Primero se introducirá la dirección, siendo 0=NORTE, 1=SUR, 2=ESTE y 3=OESTE.
 - Luego se introducirá el contenido de la casilla, siendo 0=NADA, 1=PAN_ÉLFICO, 2=FRUTO_ZARZA y 3=ORCO.
- Se puede presuponer que todos los datos que se especifican durante los pasos anteriores serán correctos, por lo que no es necesario realizar control de errores durante esta fase.

Una vez pedidos los datos, el programa realizará lo siguiente:

- Si la posición $[1,1]$ y la $[N,M]$ no están vacías, el programa mostrará un error y terminará.
- Comenzará el recorrido. El inicio siempre es en la casilla $[1,1]$, y a partir de ahí los movimientos se realizarán siguiendo las direcciones almacenadas en cada una de las casillas, menos en los casos comentados anteriormente, que debido al mareo se realizarán en caso contrario. Si alguna casilla indica una posición que no puede visitarse (por ejemplo, una casilla de la primera fila indica que hay que moverse al norte, o una de la última columna indica que hay que moverse al este), entonces el movimiento se realizará a la siguiente posición según indiquen las agujas del reloj (si no es posible moverse al norte nos movemos al este, si no es posible al este nos movemos al sur, etc). El recorrido se prolongará hasta que una de las condiciones de terminación mencionadas (Frodo muerto o llegada al final del recorrido) se alcance. Una vez que el recorrido finalice, el programa mostrará un mensaje indicando el resultado del recorrido (llegada al objetivo o muerte de Frodo y el motivo de la misma).
- Cuando pasamos a una casilla, se descontará la correspondiente energía de Frodo según la nueva casilla en la que estemos. Las casillas $[1,1]$ y $[N,M]$ son especiales y no restan energía.
- Durante el recorrido, el programa ha de guardar en un fichero de texto llamado `recorrido.txt` un registro de todos los movimientos realizados hasta el fin del recorrido. El fichero tendrá que tener el formato que se indica a continuación de forma estricta. Para cada casilla visitada, se incluirá una línea de texto en el fichero con el siguiente formato:

```
[fila,columna] - DIRECCIÓN - ENERGÍA ACUMULADA
```

Siempre hay que incluir la primera y última casilla. La dirección es la que hay en la casilla (no tiene por qué coincidir con la dirección tomada). Antes y después de cada guión hay un espacio en blanco. Un ejemplo de fichero podría ser:

```
[1,1] - ESTE - 25
[1,2] - SUR - 24
[2,2] - ESTE - 26
...
```

Como se puede observar, en este ejemplo se ve que en la casilla $[1,2]$ no había nada (pierde una unidad de energía), pero en la $[2,2]$ comió pan élfico, ya que recuperó 2 unidades de energía.

¹ Pedir los datos sin pasos intermedios significa que entre la introducción de un determinado valor y la introducción del siguiente no tiene que existir ninguna acción que tenga que realizar el usuario (preguntar si se quiere continuar o no, confirmaciones, mensajes que esperen la pulsación de una tecla, etc...)

Observaciones

- Es obligatorio hacer control de errores de entrada/salida. No se dará apta ninguna práctica que de fallos en tiempo de ejecución por motivos de trabajo con ficheros.
- El tamaño máximo tanto de N como de M será de 500 (la matriz máxima será 500x500).
- El formato de los ficheros con las casillas visitadas ha de seguirse de forma muy estricta, ya que la corrección se realizará comparando ficheros de soluciones con los que genere la práctica. Si no coinciden, se entenderá que el programa no funciona correctamente.
- Se prohíbe el uso de las funciones: `Delay`, `GotoXY`, `WhereX`, `WhereY`, `TextColor`, `TextBackground`, `Sound`, `NoSound`. También se prohíbe usar la unit `Graph`.
- Pista: con las estructuras de datos adecuadas la práctica es una cosa muy sencilla. Sin ellas, probablemente escribiréis un programa innecesariamente largo y complejo. Ir pensando en diseñar buenas estructuras de datos para representar el recorrido, representar las posiciones, para almacenar la ruta, representar a Frodo, etc.
- Es obligatorio que el código esté identado y comentado. Para cada función y procedimiento han de figurar los siguientes datos en los comentarios:
 - Descripción del valor de retorno (en caso de funciones).
 - Descripción de los parámetros de la función o procedimiento (si son de entrada o salida y para qué se utilizan).
 - Pequeña descripción de lo que hace la función o el procedimiento.

Normas de entrega

- El programa debe implementarse utilizando el compilador Free Pascal versión 2.0.2 (no sirve otra versión) en plataforma Linux. Para la corrección se compilará el código fuente y posteriormente se ejecutará el programa. Dicha compilación/ejecución se realizará en una máquina con Linux y el compilador Free Pascal. Si el alumno utilizara otro compilador o sistema operativo para desarrollar el programa es responsabilidad suya el comprobar que funciona bajo las condiciones mencionadas.
 - Se puede realizar la práctica individualmente o en grupos de dos personas. En caso de hacerlo en grupo se entregará una práctica por grupo.
 - La entrega de las prácticas ha de seguir las siguientes normas:
 - Hay que entregar código fuente y ejecutable para sistemas Linux.
 - Al hacer login con vuestros datos de alta en la asignatura, encontraréis en vuestra carpeta de usuario una carpeta llamada "EntregaPracticaViaje" (no tenéis que crearla). Debéis dejar ahí los ficheros antes de la fecha indicada ya que posteriormente no podréis escribir en ella.
 - Hay que enviar un correo electrónico a la dirección carlos.testera@unileon.es sin contenido en el cuerpo del mensaje (y sin enviar la práctica como adjunto).
El asunto del correo tiene que ser de la forma: << Práctica viaje – [Iniciales – dni] >>
Ej.--> Práctica viaje – [Abc – 71223344Z]
- Si la práctica la entregan dos personas se pondrán los datos seguidos. Un ejemplo sería:
Ej. --> Práctica viaje – [Abc – 71223344Z][Hjk – 71998877A]
- (IMPORTANTE: Es fundamental que respetéis el formato del asunto tal y como se muestra, ya que se usará un filtro para desviar los mensajes que contienen las prácticas. Los que no cumplan dicho formato no serán detectados y no se tendrán en cuenta).

La fecha límite de entrega es el 25 de Enero de 2007