

Práctica 2. Programación dinámica

21 de noviembre de 2017

Como ya se comentó en el enunciado general de las prácticas, no siempre se tienen suficientes recursos (*ases*) para enviar al planeta asaltado todas las defensas disponibles. Esta práctica consiste en diseñar un algoritmo para la selección de las defensas de la base de entre todas las disponibles. Debe ser un algoritmo determinista, esto es, para una misma configuración de entrada debe devolver siempre la misma salida.

Antes de proceder a la creación de dicho algoritmo se explicará cómo debe generarse la biblioteca dinámica donde se alojará el algoritmo. Sitúese en el directorio BASE/p1 y ejecute la siguiente orden.

```
cp libDefenseStrategy.default.so libDefenseStrategy.so
```

La orden anterior hará el simulador utilice la estrategia de ubicación de las defensas por defecto. Puede utilizar cualquier otra estrategia que considere oportuna para desarrollar sus prácticas, pero esta será la estrategia defensiva que se utilizará para evaluar el rendimiento de su segunda práctica. A continuación sitúese en el directorio BASE/p2 y ejecute la siguiente orden.

```
make test
```

Compruebe que el simulador se ejecuta correctamente y consulte el fichero *Makefile* para entender la orden que acaba de ejecutar.

La estrategia para la selección de las defensas de la base se encuentra implementada en el fichero *libDefensesSelection.so*. Antes de sobrescribir este fichero es conveniente realizar una copia del mismo, de tal forma que pueda volver a recuperarse este comportamiento en cualquier momento. Ejecute la siguiente orden.

```
cp libDefensesSelection.so libDefensesSelection.default.so
```

Para recuperar de nuevo esta estrategia defensiva solo tendrá que ejecutar las siguientes órdenes.

```
rm libDefensesSelection.so
cp libDefensesSelection.default.so libDefensesSelection.so
```

Cree una copia del fichero *DefensesSelection.example.cpp* ejecutando la siguiente orden.

```
cp DefensesSelection.example.cpp DefensesSelection.cpp
```

Abra el fichero de nombre *DefensesSelection.cpp* con algún editor de texto y localice la función *selectDefenses*. Deberá implementar su versión del algoritmo dentro de dicha función. Puede añadir al fichero *DefensesSelection.cpp* tantas funciones auxiliares como considere necesarias pero deben estar contenidas siempre dentro del fichero *DefensesSelection.cpp*. Ejecute la siguiente orden para generar la nueva versión de la biblioteca.

```
make
```

La memoria de la práctica deberá realizarse en latex. Para simplificar esta tarea se han creado una serie de documentos de texto y un *script* que se encarga de generar la memoria a partir de ellos. Usted solo deberá editar los documentos de texto correspondientes a cada ejercicio (*e1.tex*, *e2.tex*...). Abra el fichero *e1.tex* en un editor de texto básico y modifique ligeramente su contenido. Ejecute la siguiente orden y compruebe el resultado.

```
make doc
```

Si ha seguido atentamente las instrucciones anteriores, el empaquetado de la memoria para su entrega a través del campus virtual es muy sencillo. Basta con que ejecute la siguiente orden.

```
make pack
```

Renombre el fichero *APELLIDO1_APELLIDO2_NOMBRE-p2.tar.gz* para hacerlo coincidir con su nombre y apellidos.

1. Ejercicios

1. Formalice a continuación y describa la función que asigna un determinado valor a cada uno de los tipos de defensas.
2. Describa la estructura o estructuras necesarias para representar la tabla de subproblemas resueltos.
3. En base a los dos ejercicios anteriores, diseñe un algoritmo que determine el máximo beneficio posible a obtener dada una combinación de defensas y *ases* disponibles. Muestre a continuación el código relevante.
4. Diseñe un algoritmo que recupere la combinación óptima de defensas a partir del contenido de la tabla de subproblemas resueltos. Muestre a continuación el código relevante.

Todos los ejercicios tienen la misma puntuación. No es necesario que explique el código fuente en los ejercicios en los que se le solicita que lo incluya. Puede incrustar en el código los comentarios que considere oportunos.