

Grabación de programas

Objetivos

- Diseñar un algoritmo basado en alguna de las estrategias de diseño de algoritmos vistas en el temario para resolver un problema de la forma más eficiente posible.
- Saber explicar el funcionamiento de un algoritmo de forma que sea entendible por una persona "ajena" a la asignatura.
- Saber escribir el pseudocódigo de un algoritmo.
- Implementar un algoritmo en C o C++ a partir de su pseudocódigo y verificar su correcto funcionamiento.
- Calcular la complejidad temporal asintótica de un algoritmo.
- Manejar la bibliografía fundamental de la materia, así como las diferentes fuentes y recursos on-line.

Enunciado

Disponemos de un conjunto de programas que queremos grabar en una memoria. Se quiere implementar un programa que nos diga qué programas hay que grabar en la memoria para

- 1.- Grabar el mayor número de programas.
- 2.- Grabar ocupando el máximo espacio en la memoria.

Un programa solamente se graba en la memoria si se pueden grabar todos los ficheros que contiene. En caso contrario, no se graba el programa en la memoria. La capacidad de almacenamiento de la memoria USB y el espacio que ocupan los programas vienen dados en GB.

Actividades

Actividad 1: Memoria

Se realizará una memoria que contendrá:

- **PORTADA:**

Una hoja con el nombre de la asignatura, el nombre de la práctica ("Grabación de programas"), el curso académico (2020/2021), y los nombres y apellidos de los componentes del grupo.

- **ÍNDICE O TABLA DE CONTENIDO**

La siguiente página contendrá el índice del trabajo. En el índice se incluirán los apartados descritos a continuación **manteniendo los nombres y la numeración** y se incluirá el número de página donde comienzan.

1.- Algoritmo para grabar el mayor número de programas

1.1.- Descripción y tipificación

Descripción detallada del funcionamiento de un algoritmo lo más eficiente posible que resuelva el problema y que se ajuste exclusivamente a lo que se pide en el enunciado. Esta descripción debe realizarse de forma que quede claro su funcionamiento y sea entendible por una persona ajena a la asignatura (por ejemplo, amistades que no estudien informática, hermanos/-as, padres,...).

Además se realizará la tipificación del problema (variables principales del problema, qué significan, restricciones, función objetivo).

1.2.- Estrategia de programación

Identificación de la estrategia general de programación que utiliza el algoritmo descrito en los apartados anterior y justificación.

1.3.- Pseudocódigo

Pseudocódigo del algoritmo utilizando la nomenclatura vista en la asignatura.

1.4. Ejemplo

Inclusión de un ejemplo de aplicación y traza detallada del algoritmo. El ejemplo será distinto al que se incluye en el enunciado de esta práctica.

1.5.- Complejidad asintótica

Análisis y justificación de la complejidad asintótica del algoritmo del pseudocódigo.

2.- Algoritmo para grabar el máximo espacio en la memoria.

2.1.- Descripción y tipificación

Descripción detallada del funcionamiento de un algoritmo **lo más eficiente posible** que resuelva el problema. Si se utiliza parte del Algoritmo 1 en el Algoritmo 2, en los apartados de la memoria de este segundo algoritmo se debe dedicar el esfuerzo a explicar aquellas partes que sean realmente nuevas. Es decir, no se debe copiar y pegar literalmente aquellas partes desarrolladas en el punto 1 de la memoria y volver a repetirlas aquí sin aportar nada nuevo.

2.2.- Estrategia de programación

2.3.- Pseudocódigo

2.4.- Ejemplo

2.5.- Complejidad asintótica

3.- Bibliografía

Especificación de toda la bibliografía consultada con el siguiente formato:

- | | |
|-----------------------------|--|
| • Para libros, apuntes,...: | • Para direcciones web: |
| Título: | Título: |
| Autor/-es: | Url: (enlace directo a la página web) |
| Año: | |

El formato para la memoria es:

- Títulos de los apartados: Arial, 12
- Texto general: Arial, 10
- Interlineado: Sencillo o simple.
- Párrafo: Espaciado anterior y posterior: 0
- Márgenes: 2 cm. a cada lado.
- Las páginas irán numeradas y la numeración se corresponderá con la numeración de la tabla de contenido que se indica en el índice.

Actividad 2: Programa

Implementación de un programa en C/C++ correspondiente al pseudocódigo de los algoritmos de la memoria (siguiendo el modo de transformación visto en la asignatura). El programa presentará un menú con las siguientes opciones:

- 1.- Obtener lista de programas.
- 2.- Capacidad disponible en la memoria.
- 3.- Grabar máximo número de programas.
- 4.- Grabar máximo de capacidad.
- 5.- Salir.

A continuación, se detalla qué debe realizar cada opción.

1.- Obtener lista de programas.

Se leerán los datos del fichero que contiene los programas. El nombre del fichero se pedirá por pantalla. Este fichero es un fichero de texto sin formato (con extensión .txt) que contendrá en la primera línea el número total de programas y en el resto de líneas el nombre del programa y el tamaño en GB de cada uno de los programas. Por ejemplo el siguiente fichero contiene la información para un total de 4 programas con los nombres que se indican y los tamaños son 1.2, 0.5, 2 y 0.7 GB, respectivamente.

```
4
Office
1.2
Paint
0.5
Autocad
2
Vs-Code
0.7
```

2.- Capacidad disponible de la memoria.

Se pedirá cuál es el espacio libre en la memoria en GB.

3.- Grabar máximo número de programas.

4.- Grabar máximo de capacidad.

Se obtendrá un listado con los programas que se graban en el disco (número de programa) y su tamaño, según la opción correspondiente. Al final del listado se imprimirá el número total de programas que se pueden grabar y cuánto ocupan.

5.- Salir.

Con esta opción se saldrá del programa. Mientras no se pulse esta opción se volverá a presentar el menú del programa para que se pueda elegir cualquiera de las opciones del mismo.

Modo de entrega

La práctica se realizará **en grupos de 2 o 3 componentes**.

Se dispondrá de una tarea en el campos virtual de la asignatura, dentro del acceso identificado, llamada "Convocatoria Ordinaria (opción 1 – evaluación continua): Proyecto". Uno de los componentes del grupo entregará un archivo comprimido llamado **projectompa2021_evcontinua.zip**. Este archivo **solamente** contendrá los siguientes ficheros: grabar.pdf (memoria de la práctica), grabar.cpp (código fuente del programa) y grabar.exe (ejecutable). Se indicará el nombre de los miembros del grupo tanto en el fichero .pdf como en el .cpp.

Fecha fin de entrega: 14 de Junio a las 23:59.

Evaluación

Esta práctica constituye 2.0 puntos de la nota final de la asignatura. La distribución de la puntuación es la siguiente:

Algoritmo máximo número de programas (1.0 puntos)

Algoritmo máxima capacidad (1.0 puntos)

La realización de un algoritmo comprende el desarrollo de la memoria y del programa.

Se valorará:

- Memoria
 - Desarrollo correcto de los apartados especificados en la actividad 1.
 - Algoritmo propuesto para la resolución del problema. Se valora que sea un algoritmo lo más eficiente posible.
 - Pseudocódigo según la notación vista de clase.
 - Claridad en la redacción, sin errores de sintaxis (faltas de ortografía, tildes,...).
- Programa
 - El correcto funcionamiento del programa.
 - El formato de salida por pantalla, claridad y facilidad de manejo del programa.
 - Adecuación de la implementación del algoritmo con el pseudocódigo de la memoria.
- Entrevista personal

Se podrá realizar una entrevista individual a los componentes del grupo donde el profesorado realizará preguntas sobre la práctica, de forma que las respuestas del alumno serán decisivas para la nota final del alumno en la práctica, pudiendo haber diferentes puntuaciones entre los miembros de un grupo según esta entrevista.

IMPORTANTE:

La nota de la práctica será un 0 si:

- Falta algún apartado de la memoria de la opción entregada.
- Se utiliza pseudocódigo que no se corresponde con el visto en la asignatura.
- Falta el programa que implemente el algoritmo/-s descritos en la memoria.
- El programa implementado no se corresponde con el pseudocódigo descrito en la memoria.
- El código del programa da errores de compilación y/o advertencias (warnings), utilizando la versión del compilador minGW para C/C++ instalada en el aula de prácticas.
- El programa no obtiene correctamente la solución del problema.
- Los nombres de los ficheros entregados no se ajustan a los nombres especificados en el apartado "Modo de entrega" y/o se incluyen más ficheros que los especificados.
- Se detectan indicios de copia entre prácticas. En este caso la nota de **TODAS las prácticas implicadas** será un 0. No se contempla ningún tipo de justificaciones: "Hemos comentado cosas de la práctica entre nosotros,...", "la práctica la he hecho yo y él/ella se la ha copiado", "no sé cómo ha llegado la práctica a manos de otra persona",....)
- El alumnado no defiende adecuadamente la práctica en la entrevista personal.

!!! COMPROBAD ESTOS PUNTOS ANTES DE ENTREGAR LA PRÁCTICA !!!