



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
ESTRUCTURAS DE DATOS Y ALGORITMOS I
M. en I. Fco. Javier Rodríguez García



Grupo

Examen Final	
Alumno:	
Número de cuenta:	
Fecha:	
Calificación:	

1 Objetivos

Que el alumno demuestre los conocimientos adquiridos durante el semestre en la solución de un problema clásico.

Conocimientos previos:

- Estructuras de datos
- Algoritmos
- Lenguaje C
- Tipos de datos definidos por el usuario
- Programación por capas

2 Ejercicios

2.1 Problema de la mochila con programación dinámica

Resolverá el problema conocido como “De la mochila 1-0” utilizando la técnica de programación dinámica. En este problema no se permite partir a los objetos: o se meten a la mochila completos, o no se meten.

2.1.1.ADT Objeto

Escribirá un ADT Objeto con las siguientes propiedades y operaciones:

- Atributos
 - Peso
 - Valor
- Conjunto mínimo de operaciones
 - **SetValor()**, **SetPeso()**, **GetValor()**, **GetPeso()**

2.1.2.Lista inicial de objetos

Esta lista contendrá a todos los objetos (candidatos) posibles. No la confunda con la lista para el conjunto solución, porque no todos los objetos en esta lista tendrían cabida en la mochila. Dicho de otro modo, puede tener una lista de n objetos, pero solamente cabrían m objetos en la mochila (es decir, su conjunto solución tendría m elementos; con $m \leq n$). Ud. decide la representación de esta lista.

2.1.3. Conjunto solución

El conjunto solución es la lista de objetos en la mochila. Ud. decide la representación de este conjunto.

2.1.4. Programación dinámica

Escriba un algoritmo recursivo para resolver este problema. Su algoritmo deberá estar en una función llamada **MochilaPD()**. La tabla de resultados intermedios será una matriz. Ud. decida el tipo de regreso y los argumentos pertinentes. También Ud. decidirá si la matriz es estática o dinámica.

Nota: A diferencia del problema del cambio, en el problema de la mochila la tabla de resultados intermedios no se prellena, comienza completamente vacía (tome en cuenta que el valor '0' es un valor válido, por lo tanto use '-1' para indicar que una celda no ha sido calculada). Así mismo, cabe la posibilidad que mientras se está llenando obtenga valores negativos para las filas y columnas; tome en cuenta estos casos.

2.1.5. Driver program

Su driver program deberá pedirle al usuario el número inicial de objetos, el peso límite, y a continuación el valor y peso de cada objeto. Al final imprimirá el conjunto solución y la tabla de resultados intermedios.

Puede usar un arreglo **const** de objetos mientras desarrolla su programa, pero *en la versión final deberá pedirle al usuario la cantidad de objetos y los datos para cada uno de ellos.*

Nota: No está demás recordarle que no está permitido usar variables globales ni **goto**'s.

Entregables: Los códigos fuente en un archivo comprimido con nombre examen_final.zip (o .rar, o .tar.bz, o .7z, etc.)

3 Referencias y bibliografía recomendada

- <http://webdiis.unizar.es/asignaturas/EDA/ea/slides/4-Programacion%20dinamica.pdf> pp 11-17
- <http://elvex.ugr.es/decsai/algorithms/slides/6%20Dynamic%20Programming.pdf> pp 23-26 (cambio pp 11)
- [BRASSARD97]
- [WIRTH85]
- [GALVE93]

- [BAASE02]
- [WIRTH76]
- [LÓPEZ09]
- Cualquier texto sobre estructuras de datos (no importa el lenguaje de programación del que hable).
- [JOYANES05]
- [DEITELxx]
Deitel, H. M., Deitel, P. J. **Cómo programar en C/C++**. 2da. ed. ESPAÑA: McGraw-Hill, 20xx.
- Referencia del language C
<http://www.cplusplus.com/reference/>