

# Proyecto 2: El Problema de la Mochila

## Martes 4 de Setiembre

### I. DESCRIPCIÓN GENERAL

Se debe programar el algoritmo visto en clases para resolver el Problema General de la Mochila (*Knapsack*). La interfaz debe ser gráfica. Toda la programación debe realizarse en C sobre Linux. No se pueden cambiar las especificaciones de este documento.

### II. ENTRADA Y SALIDA

La interacción con el usuario se hará por medio de interfaces gráficas que deben ser desarrolladas con GTK y Glade. Debe haber un estilo uniforme en el *look & feel* de todas las interfaces. El programa tendrá la posibilidad de grabar archivos con los datos particulares de cada problema ingresado por el usuario, para que puedan ser cargados de nuevo y editados si el usuario así lo desea. El formato de este archivo queda a discreción de cada grupo de trabajo. Se espera que, el día de la revisión, cada grupo cuente con bastantes archivos de pruebas para mostrar las capacidades de su proyecto.

### III. MENÚ PRINCIPAL

En el Proyecto 0 de este curso se desarrolló un menú principal que podía “lanzar” diversos algoritmos. El algoritmo solicitado en este proyecto será implementado como un programa independiente que será ejecutado desde el menú mencionado. Se debe activar una “burbujita” con una descripción general de este algoritmo cuando el cursor se pose sobre la opción respectiva en el menú principal (*tooltip*).

### IV. PROBLEMA DE LA MOCHILA

Usando los algoritmos vistos en clase, este programa resolverá el “problema de la mochila”, en sus versiones 1/0, *bounded* y *unbounded*. Se espera que la interfaz gráfica sea lo más flexible posible.

El usuario debe proporcionar la capacidad máxima de la mochila (entre 0 y 20), la cantidad de objetos a considerar (entre 1 y 10), y para cada objeto su costo, valor y cantidad disponible (incluido el caso de que esta cantidad fuera infinito). Cuando el usuario termine la entrada, su programa mostrará el problema inicial en forma matemática.

**Trabajo extra opcional 1:** Permitir cantidades bastante mayores de objetos (más de 10) y de capacidad de la mochila (más de 20). La interfaz gráfica puede usar algún tipo de *scroll*.

La salida gráfica será la tabla final de la mochila tal como se hizo en clases, con los colores verde y rojo en los lugares apropiados. Además se debe mostrar explícitamente la solución final de manera completa.

### V. REQUISITOS INDISPENSABLES

La ausencia de uno solo de los siguientes requisitos vuelve al proyecto “no revisable” y recibe un 0 de calificación inmediata:

- Todo el código debe estar escrito en C
- El proyecto debe compilar y ejecutar en Linux
- Todas las interfaces deben ser gráficas
- Se debe usar GTK y Glade
- El programa se debe invocar desde el menú desarrollado en el Proyecto 0.
- No debe dar “Segmentation Fault” bajo ninguna circunstancia.

**Trabajo extra opcional 2:** hacer la demostración en una máquina que levante Linux de manera real (puede ser dual), es decir no usar máquinas virtuales.

### VI. FECHA DE ENTREGA

Revisiones en clase el **Martes 4 de Setiembre**. Mande además un .tgz con todo lo necesario (fuentes, makefile, readme, etc.) a [torresrojas.cursos@gmail.com](mailto:torresrojas.cursos@gmail.com). Ponga como subject: I.O. - Proyecto 2 - Fulano - Mengano - Sotano, donde Fulano, Mengano y Sotano son los 3 miembros del grupo.