

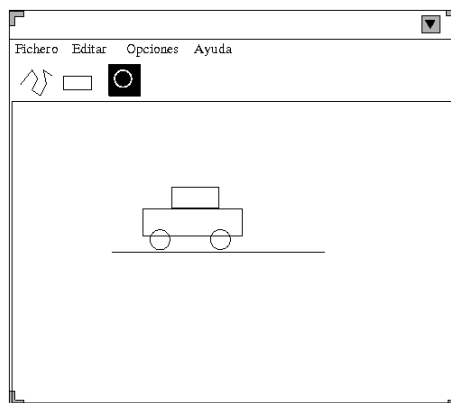


## Nuevas tecnologías de la programación

### Práctica 2: Editor gráfico en Qt (curso 2011-2012)

#### Descripción

La práctica consiste en la implementación utilizando la librería Qt (versión 4.x) de un editor gráfico de dibujo vectorial al estilo del famoso programa *xfig* de linux, aunque bastante más simple. Para diseñar el interfaz gráfico puede usarse si se desea **Qt Designer**, **Qt Creator** o **QDevelop**, aunque es recomendable programar el interfaz gráfico manualmente.



#### Requerimientos mínimos

Los requerimientos mínimos que se piden al programa son los siguientes:

1. La zona del tablero (widget central) se implementará creando una subclase de la clase `QWidget`, dónde deben controlarse los eventos de ratón y eventos *paint* para que no se pierdan los contenidos del tablero, con este evento.
2. Incluir una barra de menús, que contenga obligatoriamente el menú **Fichero** (el resto de menús son opcionales) con al menos las siguientes opciones:
  - **Nuevo:** Borra el dibujo actual del tablero.
  - **Salir:** Para salir del programa.
3. El programa debe permitir introducir al menos objetos línea y objetos rectángulo. Para ello podría usarse por ejemplo el siguiente método:
  - **Líneas:** Pinchar con el botón izquierdo del ratón en la posición inicial y posteriormente en la posición final.
  - **Rectángulos:** Pinchar con el botón izquierdo del ratón en la posición del vértice superior izquierdo del rectángulo y luego en la posición del vértice inferior derecho.



4. Habrá botones (línea y rectángulo) para seleccionar el modo actual. Si por ejemplo pinchamos en el de **líneas** el editor pasa a modo *dibujar líneas* y a partir de ahora se comienzan a introducir líneas al ir pinchando con el ratón en la zona de dibujo.

Opcionalmente se pueden incluir en el programa otras mejoras, explicándolas en la documentación del programa. Podéis mirar por ejemplo el programa *xfig* para tomar ideas de qué mejoras introducir. Por ejemplo algunas mejoras podrían ser las siguientes:

- Dibujar los distintos objetos mediante el uso de líneas elásticas en lugar de con el método explicado en el punto 3 de los requisitos mínimos. Después de pinchar con el ratón en el primer punto del objeto el objeto se iría dibujando al arrastrar el ratón, y quedará fijo al soltar el botón del ratón.
- Permitir introducir otros objetos como círculos, elipses, polilíneas, etc.
- Uso de barras de utilidades (toolbars), por ejemplo para incluir los botones de tipo de objeto a dibujar.
- Permitir introducir texto.
- Mover, borrar y copiar objetos.
- Poder seleccionar un vértice de un objeto para escalarlo o bien moverlo.
- Opciones de *Deshacer* (Undo) y *Rehacer*.
- Poder seleccionar ciertas características de la línea como si es continua o discontinua, grosor, color, etc.
- Edición de objetos (cambio de sus características).
- Salvar y cargar el dibujo en un fichero.
- Incluir el menú Opciones, con opciones como selección del color para los objetos que se dibujen a continuación, etc.
- Incluir un menú de Ayuda.
- Uso de diálogos de confirmación: por ejemplo, para confirmar si queremos borrar el dibujo actual, etc.
- Uso de diálogos para cargar o salvar (en caso de que se implemente la mejora).
- Zoom.



### Documentación a entregar

El programa y la documentación se entregará a través de la plataforma de docencia del departamento de Ciencias de la Computación e I.A. (<http://decsai.ugr.es>) enviando un fichero tar comprimido (practica2.tgz). Sólo es necesario que lo envíe uno de los autores de la práctica.

El fichero debe contener el código fuente, makefile, ejecutable para linux y cualquier otro fichero que sea necesario para compilar el programa.

**IMPORTANTE:** El fichero debe contener también un fichero `Readme.txt` (fichero ASCII) cuyo contenido sea los apellidos y nombre de cada uno de los autores de la práctica, así como la dirección e-mail de cada uno de ellos (una línea por cada autor).

El fichero también debe contener la documentación de la práctica. Tal documentación debe entregarse en los formatos postscript o bien pdf. Esta documentación contendrá:

1. Enunciado del problema.
2. Análisis del problema: requerimientos, modularidad, estructuras de datos, justificación de la solución, diseño de los algoritmos, etc.
3. Manuales de usuario y compilación del programa.

### Tiempo de realización

El tiempo estimado para la realización de la práctica es de 8 horas. O sea de 4 semanas.

### Fecha de entrega

La práctica puede ser enviada hasta el día 20 de diciembre de 2011. Al día siguiente se publicará en la página web de la asignatura (<http://decsai.ugr.es/~acu/NTP>) la lista de alumnos que han enviado la práctica.