

# Nuevas Tecnologías de la Programación

8 de Febrero de 2007

1. (1.5 pts) El objetivo del siguiente programa Xlib es mostrar una ventana en la que aparezca dibujado un cuadrado en las coordenadas 10,10 con lado igual a 30. El programa no está bien construido y al ejecutarlo no aparece nada (ni el cuadrado, ni la ventana). Explica las causas de que no aparezca la ventana y que no aparezca el cuadrado. Indica también las modificaciones que hay que hacer para que funcione el programa correctamente.

```
#include <X11/Xlib.h>
int main(int argc, char **argv){
    Display *display;
    int screen_num;
    Window win;
    XEvent report;
    display=XOpenDisplay(NULL);
    screen_num = DefaultScreen(display);
    win = XCreateSimpleWindow(display, RootWindow(display,screen_num),
        50, 100, 200, 200, 2, BlackPixel(display,screen_num),
        WhitePixel(display,screen_num));
    XMapWindow(display, win);
    XDrawRectangle(display, win, DefaultGC(display,screen_num),10,10,30,30);
}
```

2. (1.5 pts) En un programa construido con Xlib, cuando una ventana recibe un evento Expose, significa que ésta ha perdido parte de sus contenidos. Explica brevemente como las siguientes filosofías para recuperar sus contenidos:
  - a) Redibujar todos los contenidos de la ventana pero sólo cuando count==0
  - b) Usar doble buffer a través
3. (2 puntos) Explica las dos filosofías existentes en **Xforms** para controlar la interacción con el usuario escribiendo el esquema de un programa que cree un form con un botón que permita salir del programa.
4. (5 puntos) Escribe un programa en Java que muestre una ventana (**Frame**) con un botón (**Button**) que permita salir del programa. Además el frame contendrá un campo de texto (**TextField**). El programa debe crear un hilo (**Thread**) adicional para llevar el control de la hora actual, que será mostrada cada segundo en el campo de texto. Para obtener la hora actual podemos usar:

```
Date date=new Date();
Calendar cal=Calendar.getInstance();
cal.setTime(date);
int segundos=cal.get(Calendar.SECOND);
int minutos=cal.get(Calendar.MINUTE);
int horas=cal.get(Calendar.HOUR);
```

El método `setText(String)` de `TextField` permite poner un `String` en el campo de texto. El evento de pulsación en el botón se controla con el interfaz `ActionListener` implementando el método:

```
void actionPerformed(ActionEvent event).
```

Añadir la posibilidad de definir una hora en la que se mostrará el mensaje "Tienes una cita", en una ventana de diálogo (objeto `Dialog` o `JDialog` que contendrá un `Label` o `JLabel` para el mensaje y un botón para hacer desaparecer el diálogo). Cuando el reloj alcance la hora definida, se mostrará el diálogo con el mensaje y el botón.

# Nuevas Tecnologías de la Programación

8 de Febrero de 2007

1. (1.5 pts) El objetivo del siguiente programa Xlib es mostrar una ventana en la que aparezca dibujado un cuadrado en las coordenadas 10,10 con lado igual a 30. El programa no está bien construido y al ejecutarlo no aparece nada (ni el cuadrado, ni la ventana). Explica las causas de que no aparezca la ventana y que no aparezca el cuadrado. Indica también las modificaciones que hay que hacer para que funcione el programa correctamente.

```
#include <X11/Xlib.h>
int main(int argc, char **argv){
    Display *display;
    int screen_num;
    Window win;
    XEvent report;
    display=XOpenDisplay(NULL);
    screen_num = DefaultScreen(display);
    win = XCreateSimpleWindow(display, RootWindow(display,screen_num),
        50, 100, 200, 200, 2, BlackPixel(display,screen_num),
        WhitePixel(display,screen_num));
    XMapWindow(display, win);
    XDrawRectangle(display, win, DefaultGC(display,screen_num),10,10,30,30);
}
```

2. (1.5 pts) En un programa construido con Xlib, cuando una ventana recibe un evento Expose, significa que ésta ha perdido parte de sus contenidos. Explica brevemente como las siguientes filosofías para recuperar sus contenidos:
  - a) Redibujar todos los contenidos de la ventana pero sólo cuando count==0
  - b) Usar doble buffer a través
3. (2 puntos) Explica las dos filosofías existentes en **Xforms** para controlar la interacción con el usuario escribiendo el esquema de un programa que cree un form con un botón que permita salir del programa.
4. (5 puntos) Escribe un programa en Java que muestre una ventana (**Frame**) con un botón (**Button**) que permita salir del programa. Además el frame contendrá un campo de texto (**TextField**). El programa debe crear un hilo (**Thread**) adicional para llevar el control de la hora actual, que será mostrada cada segundo en el campo de texto. Para obtener la hora actual podemos usar:

```
Date date=new Date();
Calendar cal=Calendar.getInstance();
cal.setTime(date);
int segundos=cal.get(Calendar.SECOND);
int minutos=cal.get(Calendar.MINUTE);
int horas=cal.get(Calendar.HOUR);
```

El método `setText(String)` de `TextField` permite poner un `String` en el campo de texto. El evento de pulsación en el botón se controla con el interfaz `ActionListener` implementando el método:

```
void actionPerformed(ActionEvent event).
```

Añadir la posibilidad de definir una hora en la que se mostrará el mensaje "Tienes una cita", en una ventana de diálogo (objeto `Dialog` o `JDialog` que contendrá un `Label` o `JLabel` para el mensaje y un botón para hacer desaparecer el diálogo). Cuando el reloj alcance la hora definida, se mostrará el diálogo con el mensaje y el botón.