

**Министерство образования и науки Российской Федерации**

**Федеральное агентство по образованию**

**Государственное образовательное учреждение высшего профессионального образования**

**«Московский государственный технический университет имени Н.Э. Баумана»**

**(МГТУ им. Н.Э. Баумана)**

**Факультет «Робототехника и комплексная автоматизация» (РК)**

**Кафедра «Системы автоматизированного проектирования» (РК6)**

****

**Отчет по лабораторной работе №1 по курсу**

**«Программирование графических приложений»**

**Студент: Сергеева Диана**

**Группа:** РК6-46Б

**Преподаватель:** Волосатова Т.М.

Проверил:

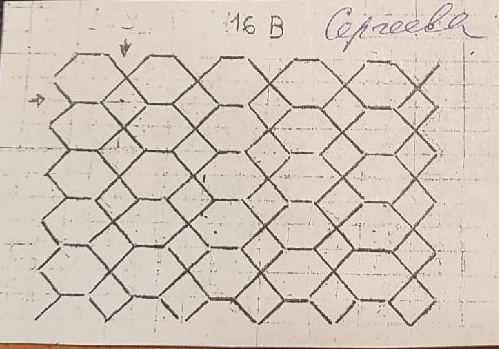
Дата:

2021 год

**16 вариант**

**Задание:**

Разработать программу для создания графического окна, фон которого отображает заданный рисунок из периодически повторяющихся геометрических фигур. Их взаимное расположение должно образовать замощение плоскости графического окна независимо от его геометрических размеров. Программную реализацию такого замощения должна обеспечивать карта пикселей минимального по размерам прямоугольного фрагмента рисунка фона, который повторяется в нем. Такая пиксельная карта должна конструироваться в программе минимальным набором примитивов графического вывода и адресоваться графическому окну структурой его атрибутов. Это должно гарантировать автоматическое тиражирование карты пикселей слева-направо и сверху-вниз в пределах графического окна, пока оно отображено на экране. Кроме того, необходимо предусмотреть возможность интерактивно инвертировать цвет рисунка фона по нажатию и отпусканию любой кнопки мыши, когда ее курсор находится в границах графического окна. Для этого в программе нужно реализовать соответствующую обработку событий и изображений для ее графического окна. При разработке программы необходимо использовать библиотечные функции базисного программного интерфейса XWindowSystem.



**Основные моменты программной реализации:**

* Программную реализацию такого замощения должна обеспечивать карта пикселей минимального по размерам прямоугольного фрагмента рисунка фона, который повторяется в нем.
* Пиксельная карта должна конструироваться в программе минимальным набором примитивов графического вывода и адресоваться графическому окну структурой его атрибутов. Это должно гарантировать автоматическое тиражирование карты пикселей слева направо и сверху вниз в пределах графического окна, пока оно отображено на экране.
* Кроме того, необходимо предусмотреть возможность интерактивно инвертировать цвет рисунка фона по нажатию и отпусканию любой кнопки мыши, когда ее курсор находится в границах графического окна. Для этого в программе нужно реализовать соответствующую обработку событий и изображений для ее графического окна. При разработке программы необходимо использовать библиотечные функции базисного программного интерфейса X Window System.
* Замощение плоскости (полу)правильными топологическими многоугольниками изучает геометрическая теория графов. На практике может быть построено бесконечно много их различных вариантов путем геометрической деформации разнообразных многоугольных элементов

**Основная идея программной генерации фона графического окна Х-клиента:**

* подготовить пиксельную карту рисунка повторяющегося элемента фона и присвоить ее идентификатор фоновому атрибуту графического окна.
* создать и визуализировать графическое окно.
* в окне гарантируется автоматическое тиражирование заданного рисунка независимо от его исходных и текущих габаритов.
* листинг содержит исходный код программы генерации фона графического окна из сопряженных многоугольников, узор из которых показан на рисунке выше.

**Код программы:**

/\* Chain Arm Ring background \*/

#include <X11/Xlib.h>

#define WIDTH 128 /\* repeat element width \*/

#define HEIGHT 64 /\* repeat element height \*/

int main(int argc, char\* argv[]) {

Display\* dpy; /\* display structure \*/

int src; /\* screen number \*/

GC gc; /\* graphic context \*/

int depth; /\* bit per color pixel \*/

Window root; /\* screen root window \*/

Pixmap pix; /\* Pixel map for ring \*/

Window win; /\* Main window \*/

XSetWindowAttributes attr; /\* window attributes \*/

unsigned long mask; /\* event mask \*/

XEvent event; /\* Event structure \*/

unsigned int done = 0; /\* exit code \*/

/\* X init \*/

dpy = XOpenDisplay(NULL);

src = DefaultScreen(dpy);

depth = DefaultDepth(dpy, src);

root = DefaultRootWindow(dpy);

gc = DefaultGC(dpy, src);

/\* Pixmap block \*/

pix = XCreatePixmap(dpy, root, WIDTH, HEIGHT, depth);

XSetBackground(dpy, gc, WhitePixel(dpy, src));

XSetForeground(dpy, gc, WhitePixel(dpy, src));

XFillRectangle(dpy, pix, gc, 0, 0, WIDTH, HEIGHT);

XSetForeground(dpy, gc, BlackPixel(dpy, src));

XDrawLine(dpy, pix, gc, 0, 32, 32, 0);

XDrawLine(dpy, pix, gc, 0, 32, 32, 64);

XDrawLine(dpy, pix, gc, 32, 63, 96, 63);

XDrawLine(dpy, pix, gc, 96, 63, 127, 32);

XDrawLine(dpy, pix, gc, 128, 32, 96, 0);

/\* Window block \*/

mask = (CWOverrideRedirect | CWBackPixmap);

attr.override\_redirect = False;

attr.background\_pixmap = pix;

win = XCreateWindow(dpy, root, 0, 0, 800, 600, 1, depth,

InputOutput, CopyFromParent, mask, &attr);

mask = (ButtonPressMask | ButtonReleaseMask |

EnterWindowMask | LeaveWindowMask |

KeyPressMask);

XSelectInput(dpy, win, mask);

XMapRaised(dpy, win);

XSetFunction(dpy, gc, GXinvert); /\* XSetFunction(dpy, gc, GXcopyInverted); \*/

/\* Dispatch block \*/

while(done == 0) {

XNextEvent(dpy, &event);

switch(event.type) {

case EnterNotify:

case LeaveNotify:

case ButtonPress:

case ButtonRelease: XCopyArea(dpy, pix, pix, gc, 0, 0, WIDTH, HEIGHT, 0, 0);

XSetWindowBackgroundPixmap(dpy, win, pix);

XClearWindow(dpy, win);

break;

case KeyPress: done = event.xkey.keycode;

break;

default: break;

} /\* switch \*/

} /\* while \*/

/\* X-Exit block \*/

XFreePixmap(dpy, pix);

XDestroyWindow(dpy, win);

XCloseDisplay(dpy);

return(done);

}

**Описание программы.**

**Графическая инициализация:**

• по запросу XOpenDisplay адресуется структура дисплея.

• Затем устанавливается по умолчанию номер экрана, его корневое окно и число цветовых плоскостей, а также графический контекст изображения.

• Для получения этих дисплейных умолчаний вызываются соответствующие дисплейные макросы DefaultScreen, DefaultRootWindow, DefaultDepth и DefaultGC.

**Формирование пиксельной карты фона:**

• формируется пиксельная карта повторяющегося элемента фона. При этом сначала по запросу XCreatePixmap создается пустая пиксельная карта требуемого размера пикселей.

• она заполняется белым квадратом по запросам XForeground (с белым макросом WhitePixel) и XFillRectangle.

• в ней рисуется требуемое изображения по запросам XForeground (с черным макросом BlackPixel) и XdrawArc и XDrawpoint. В общем случае для формирования изображения могут применяться различные примитивы графического вывода, например, прямоугольники и отрезки прямых линий, которые строятся по запросам XDrawRectangle(s), XDrawLine и XDrawSegments. Вместо такой векторной техники конструирования изображения в пиксельной карте могут также использоваться растровые шаблоны из битовых карт, которые формируются, например, программой графического клиента bitmap и отображаются по запросу **XCreateBitmapFromData.**

**Создание графического окна:**

• по запросу XCreateWindow создается графическое окно программы с заданным начальным размером пикселей, которое является подокном корневого окна экрана, копирует его визуальный класс и глубину.

• Оно также декорируется оконным менеджером по его атрибуту override\_redirect со значением False. Фоновый атрибут окна background\_pixmap фиксирует идентификатор пиксельной карты из предыдущего блока. Это гарантирует автоматическое заполнение фона окна рядами изображений пиксельной карты, независимо от изменений габаритов окна, когда оно отображается на экране.

• После создания окна, по запросу XSelectInput устанавливается маска получения событий нажатия (отпускания) кнопок мыши и клавиш на клавиатуре, когда курсор мыши находится в габаритах окна или пересекает его рамку.

• В конце оконного блока сформированное окно отображается на экране по запросу XMapRaised, перекрывая другие окна. Графическая утилита XStoreName фиксирует заголовок окна, заданный строкой "floor", в именном свойстве WM\_NAME оконного менеджера, который отображает его в своем обрамлении окна.

• Кроме того, по запросу XSetFunction в графическом контексте устанавливается функция инвертирования изображения GXinvert или GXcopyInverted для последующих операций графического вывода в цикле обработки событий, чтобы обеспечить переключение цвета фона графического окна.

**Обработка событий:**

• реализует цикл чтения и обработки событий из очереди Х-сервера, которые установлены для графического окна программы в блоке.

• Чтение поступающих событий обеспечивает запрос XNextEvent, а альтернативы их обработки выбираются по полю типа в структуре XEvent каждого полученного события.

• В частности, для событий нажатия и отпускания любой кнопки мыши, когда ее курсор находится внутри окна, предусмотрена следующая обработка. Сначала инвертируется изображение в пиксельной карте фона путем копирования ее области саму на себя по запросу XCopyArea в графическом контексте с ранее установленной функцией инвертирования GXinvert (GXcopyInverted). Затем инвертированная таким образом пиксельная карта переустанавливается для фона графического окна по запросу XSetWindowBackgroundPixmap.

• Наконец, очистка окна по запросу XClearWindow реализует инвертирование его фонового рисунка.

• Аналогичная обработка предусматривается для события пересечения курсором рамки окна в любом направлении (EnterNotify и LeaveNotify). Итерации обработки этих мышиных событий могут продолжаться, пока значение флага завершения цикла done равно 0 (False).

• В альтернативе KeyPress обработки нажатия любой клавиши на клавиатуре (когда окно имеет фокус ввода) этому флагу присваивается значение ее физического кода из поля keycode структуры XEvent полученного события. Его величина гарантированно больше 0 для любой клавиши. Поэтому значение флага done становится положительным (True) и происходит выход из цикла обработки событий для последующего завершения программы.

**Графический выход:**

• предусмотрен типовой набор завершающих действий:

• Сначала по запросу XFreePixmap освобождается память, выделенная для пиксельной карты.

• Затем по запросу XDestroyWindow закрывается графическое окно программы.

• Наконец, запрос XCloseDisplay обеспечивает разрыв связи с X-сервером. Физический код клавиши, которая была нажата для выхода из цикла обработки событий, возвращает exit-код программы.

**Синтез исполняемого кода:**

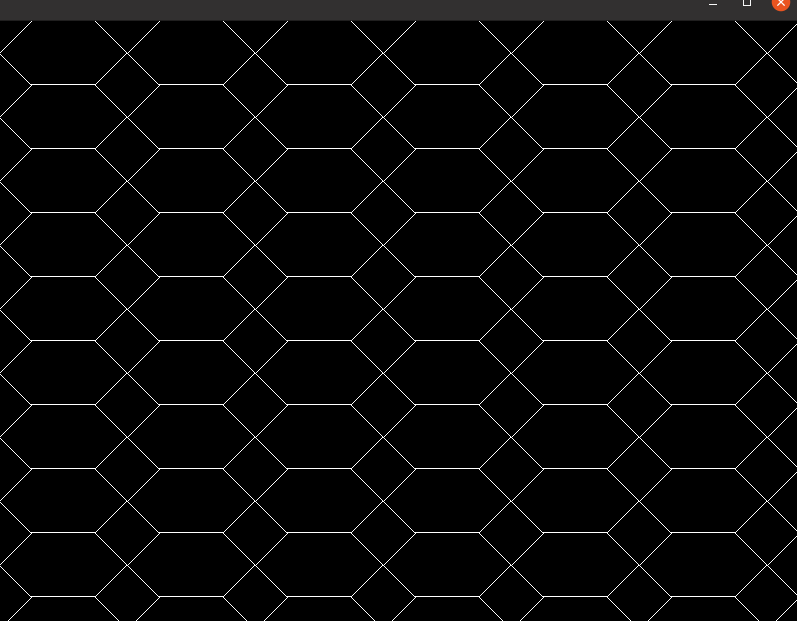
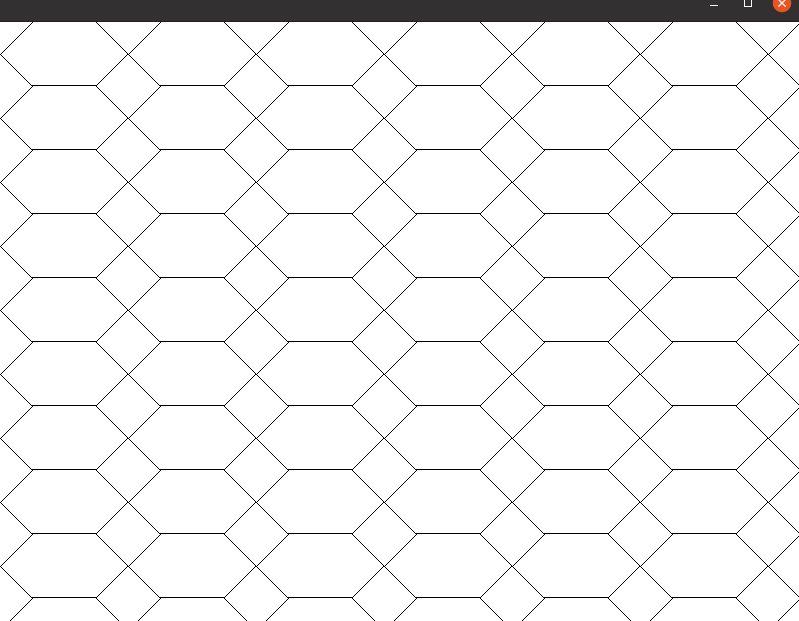
• После соответствующего редактирования исходный код программы следует сохранить под именем xlab1.c в рабочем каталоге файловой системы.

• Его компиляцию и компоновку с графической библиотекой libX11.so(.a) обеспечивает следующая командная строка, которая вводится в окне эмулятора терминала:$ cc xlab1.c −lX11

• В результате будет получен выполняемый файл a.out, который может быть вызван для отображения на экране графического окна с заданным фоном.

• Рассмотренный исходный код программы генерации фона кольчуги можно приспособить для формирования графического фона с другим узором, изменив код рисования изображения в пиксельной карте повторяющегося фрагмента фона

**Результаты работы программы:**

****

**Литература:**

1) bigor.bmstu.ru – база и генератор образовательных ресурсов

2) Лекции по курсу «Программирование графических приложений» 3) Электронный учебник: http://eufs.bmstu.ru/ee8d5d88-abfa-11e6-aa39-005056960017/09-02-2021-%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D1%8B%D0%B8\_\_%D1%83%D1%87%D0%B5%D0%B1%D0%BD%D0%B8%D0%BA\_%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5\_%D0%B2\_X\_Window\_System.htm