

Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный университет телекоммуникаций и информатики»
(СибГУТИ)

Институт информатики и вычислительной техники

09.03.01 "Информатика и вычислительная техника"
профиль "Программное обеспечение средств
вычислительной техники и автоматизированных систем"

Контрольная работа

по дисциплине «Программирование»

Выполнил(а):

Студент гр. ИП-311

_____/Подкорытова А.В./
ФИО студента

«__»_____2023 г.

Проверил:

Старший преподаватель
кафедры ПМиК

_____/Агалаков А. А./
ФИО преподавателя

«__»_____2023 г.

Оценка _____

Новосибирск 2023

Оглавление

Задание	2
Теория	2
Ход работы	2
Демонстрация работы	3
Вывод	5
Литература	5
Приложение	7

Задание

1. Сформировать на экране меню, состоящее из следующих пунктов:

1. Свободное движение объекта
2. Движение объекта с помощью клавиатуры
3. Выход из программы

2. Выбрать тип объекта и движения и реализовать пункты меню в виде подпрограмм.

Варианты свободного движения объектов.

Движение с изменяющимся углом ($dx - const$, $dy - variable$) с отражением от границ экрана.

Варианты движения объектов с помощью клавиатуры.

Управление клавишами со стрелками $\leftarrow(0,75)$, $\rightarrow(0,77)$, $\uparrow(0,72)$, $\downarrow(0,80)$.

Вид графического объекта.



Ход работы

Программа начинается с функции: `main`.

Функция `menu` – создает панель для выбора программы, вызывая её, мы должны выбрать нужное задание, это происходит благодаря смещению `y`, после чего, нажав **enter**, мы возвращаем номер задания, которое мы хотим вывести на экран, закрыв предыдущее окно.

Выбрав пункт 1, запускается функция `start()`, которая дает возможность свободно управлять направлением рисунка.

Выбрав пункт 2, запускается функция `avtomat()`, которая автоматически перемещает рисунок по полю.

Выбрав пункт 3 происходит завершение программы.

Функция `xmove(int x , int y)` - принимает на вход параметры и исходя из них рисует фигуру по нужным координатам.

Демонстрация работы

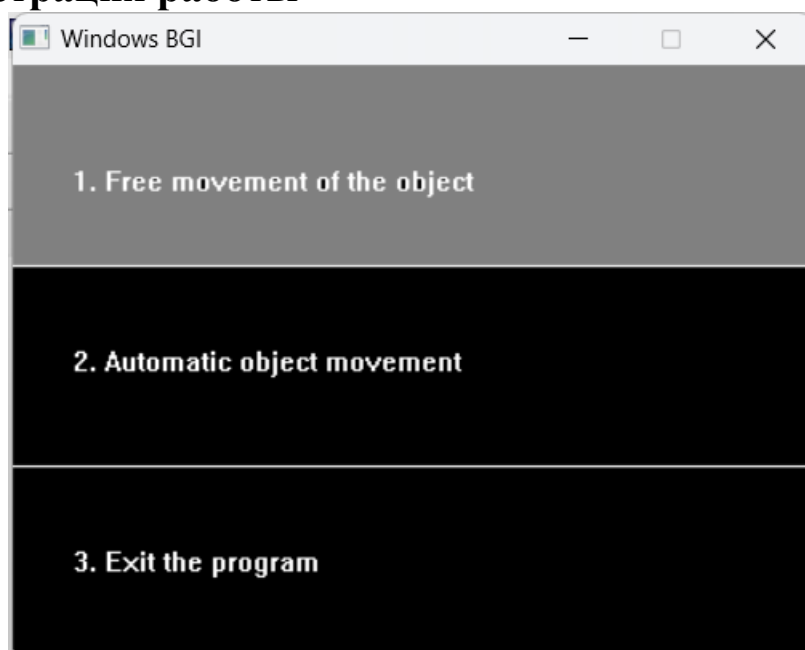


Рисунок 1. Меню программы.

В данном окне есть 3 пункта выбора, между которыми пользователь может перемещаться с помощью стрелочек вверх и вниз.

При выборе первого пункта, появляется окно с изображением, размером 1920 на 1080, в котором можно свободно с помощью стрелочек перемещать фигуру по окну(с учётом установленных границ). Если в ходе движения, нажатие на клавишу гарантирует выход за границы поля, на экране высветится уведомление, а объект останется на месте. В данном режиме есть 4 кнопки управления, соответственно движение вправо, влево и по диагонали вверх и вниз.

Чтобы выйти из данного режима необходимо нажать **enter**.

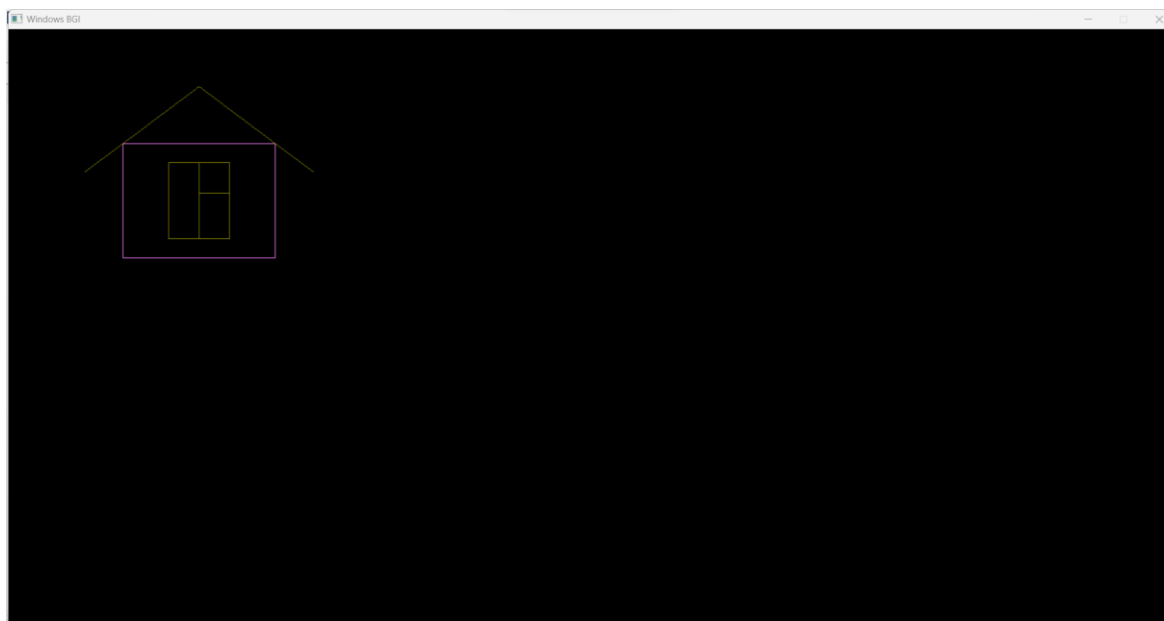


Рисунок 2. Окно программы, для самостоятельного движения объекта.

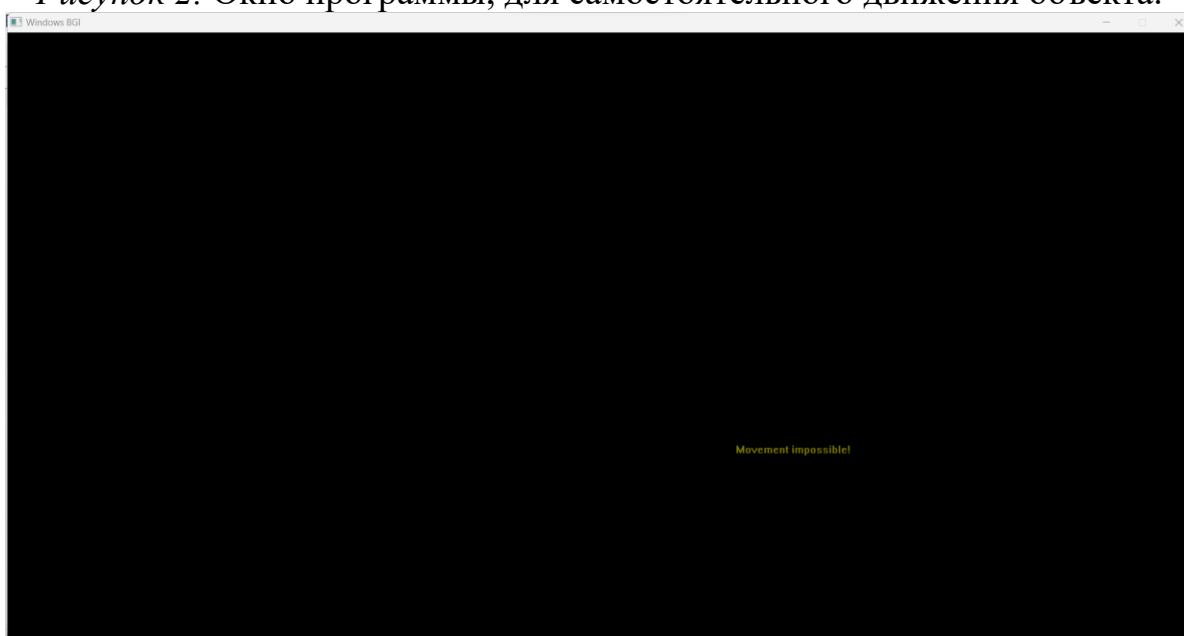


Рисунок 3. Предупреждающее сообщение о невозможности дальнейшего движения.

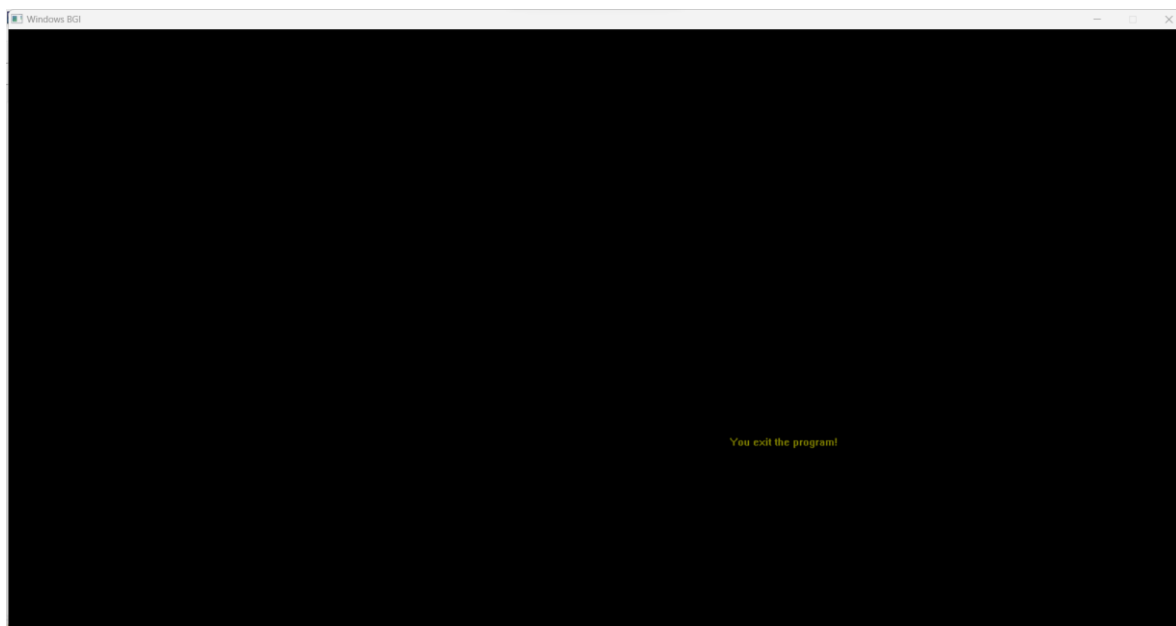


Рисунок 4. Сообщение о выходе из программы.

При выборе второго пункта начинается автоматическое движение объекта по площади окна. Это движение длится бесконечно, а завершение происходит путём выхода из окна Windows BGI.

Вывод

В ходе выполнения работы была использована библиотека "graphics.h" для программы Visual Studio. Реализовано движение объекта в среде Windows BGI с помощью клавиатуры, а также с помощью циклов, для свободного движения объектов.

Литература

Список литературы оформляется в соответствии с ГОСТ Р 7.0.5-2008 «Библиографическая ссылка. Общие требования и правила составления». Описание зарубежных источников дается на языке оригинала без перевода. При этом в названии статьи заглавной пишется только первая буква (за исключением имен собственных и символов элементов). Для описаний работ в переводимых журналах указывайте сначала русскоязычный вариант, а затем (при необходимости) вариант на иностранном языке. Адреса в Интернете рекомендуется вводить моноширинным шрифтом Courier New 11 пт или аналогичным.

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации: учебное пособие для вузов. М.: Горячая линия–Телеком, 2005. 229 с.
2. Blake I., Seroussi G., Smart N. Elliptic Curves in Cryptography. Cambridge University Press, 2002. 204 p.
3. История России: учебник / А. С. Орлов, В. А. Георгиев, Н. Г. Георгиева, Т. А. Сивохина. 2-е изд., перераб. и доп. М.: Проспект, 2004. 514 с.
4. Экономика: учебник / под ред. А. С. Булатова. 3-е изд., перераб. и доп. М.: Экономистъ, 2003. 894 с.
5. Рябко Б. Я., Фионов А. Н. Эффективный метод адаптивного арифметического кодирования для источников с большими алфавитами // Проблемы передачи информации. 1999. Т. 35, № 4. С. 1–14.
6. Лэтчфорд Е. У. С Белой армией в Сибири [Электронный ресурс] // Восточный фронт армии адмирала А. В. Колчака: [сайт]. [2004]. URL: <http://east-front.narod.ru/memo/latchford.htm> (дата обращения: 23.08.2007).

Приложение

Обязательно с новой страницы! Код программы, а также code [style!](#)

Использовать Courier New шрифт 12.

```
#include "graphics.h"
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <time.h>

int menu() //создаём окно-панель для выбора задания
{
    initwindow(400, 300);
    line(0, 100, 400, 100);
    line(0, 200, 400, 200);
    int x = 30, y = 50;
    char co = 1;
    while (co != 13)
    {
        if (y >= 300)
            y = 50;
        if (y <= 0)
            y = 250;
        outtextxy(x, 50, "1. Free movement of the object");
        outtextxy(x, 40 + 100, "2. Automatic object movement ");
        outtextxy(x, 40 + 200, "3. Exit the program");
        setfillstyle(1, 8); //заливка цветом и его цвет
        floodfill(2, y, 15);
        co = getch();
        setfillstyle(1, 0); //заливка цветом и его тип
        floodfill(2, y, 15);
        switch (co)
        {
            case 72:
                y -= 100;
```

```

        break; //вверх
    case 80:
        y += 100;
        break; //вниз
    }
}
y = (y + 50) / 100;
closegraph();
return y;
}
void xmove(int x, int y) //создание фигуры
{
    cleardevice();
    setcolor(6);
    setcolor(13);
    int x1 = 150 + x, x2 = 350 + x, y1 = 150 + y, y2 = 300 + y;
    rectangle(x1, y1, x2, y2);
    setcolor(6);
    moveto(150 + x, 150 + y);
    lineto(250 + x, 75 + y);
    lineto(350 + x, 150 + y);
    lineto(400 + x, 187 + y);
    moveto(150 + x, 150 + y);
    lineto(100 + x, 187 + y);
    moveto(175 + x, 100 + y);
    rectangle(210 + x, 175 + y, 290 + x, 275 + y);
    moveto(250 + x, 175 + y);
    lineto(250 + x, 275 + y);
    moveto(250 + x, 215 + y);
    lineto(290 + x, 215 + y);
}
void avtomat() //свободное движение
{
    int x = 0, y = 0;
    int dx = 1, dy = 1;

```

```

initwindow(1920 / 1.25, 1080 / 1.25);
xmove(0, 0);
while (1)
{
    x += dx;
    y += dy;
    if (x < 0 || x > getmaxx() - 400)
    {
        dx = -dx;
    }
    if (y < 0 || y > getmaxy() - 350)
    {
        dy = -dy;
    }
    cleardevice();
    xmove(x, y);
}
closegraph();
}

void start() //управление с клавиатуры
{
    initwindow(1920 / 1.25, 1080 / 1.25);
    xmove(0, 0);
    bool exit = false;
    int x = 0, y = 0;
    int k = 1;
    while (!exit)
    {
        int s = getch();
        switch (s)
        {
            case 75:
                if (x > 0)
                {
                    x -= 137;

```

```

        xmove(x, y);
        k = -1;
    }
    else
    {
        cleardevice();
        outtextxy(940, 530, "Movement is only possible to
the right!");
        delay(3000);
        xmove(0, y);
    }
    break;
case 77:
    if (x < 1370)
    {
        x += 137;
        xmove(x, y);
        k = 1;
    }
    else
    {
        cleardevice();
        outtextxy(940, 530, "Movement is only possible to
the left!");
        delay(3000);
        xmove(1370, y);
    }
    break;
case 72:
    if ((x == 1370) && (k == 1) && (y > 0))
    {
        x -= 137;
        y -= 78;
        k = -1;
        xmove(x, y);
    }

```

```

else if ((x == 0) && (k == -1) && (y > 0))
{
    x += 137;
    y -= 78;
    k = 1;
    xmove(x, y);
}
else if ((x > 0) && (k == -1) && (y > 0))
{
    x -= 137;
    y -= 78;
    xmove(x, y);
    k = -1;
}
else if ((x > 0) && (k == 1) && (y > 0))
{
    x += 137;
    y -= 78;
    xmove(x, y);
    k = 1;
}
else
{
    cleardevice();
    outtextxy(940, 530, "Movement impossible!");
    delay(3000);
    xmove(x, y);
}
break;
case 80:
if ((k == -1) && (y < 702) && (x == 0))
{
    x += 137;
    y += 78;
    xmove(x, y);

```

```

        k = 1;
    }
    else if ((k == 1) && (y < 702) && (x == 1370))
    {
        x -= 137;
        y += 78;
        xmove(x, y);
        k = -1;
    }
    else if ((x < 1370) && (k == -1) && (y < 702) && (x !=
        0))
    {
        x -= 137;
        y += 78;
        xmove(x, y);
        k = -1;
    }
    else if ((x < 1370) && (k == 1) && (y < 702))
    {
        x += 137;
        y += 78;
        xmove(x, y);
        k = 1;
    }
    else
    {
        cleardevice();
        outtextxy(940, 530, "Movement impossible!");
        delay(3000);
        xmove(x, y);
    }
    break;
case 13:
    cleardevice();
    outtextxy(940, 530, "You exit the program!");

```

```

        delay(5000);
        exit = true;
        break;
    default:
        break;
    }
}
closegraph();
}
int main()
{
    int cursor;
    cursor = menu();
    while (cursor != 3)
    {
        if (cursor == 1)
            start();
        else if (cursor == 2)
            avtomat();
        cursor = menu();
    }
    return 0;
}

```