

Основы программирования

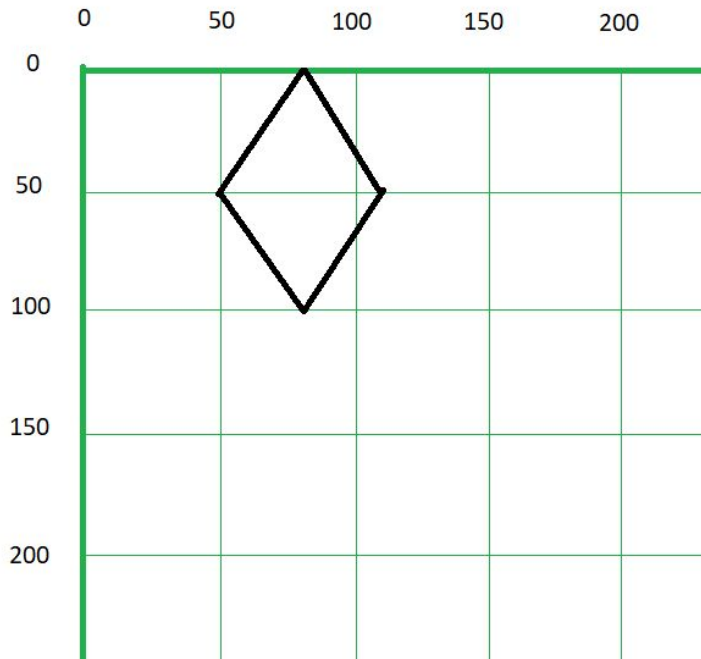
Лабораторная работа №9

WinAPI Графика – Относительные
координаты.

Функции с параметрами.

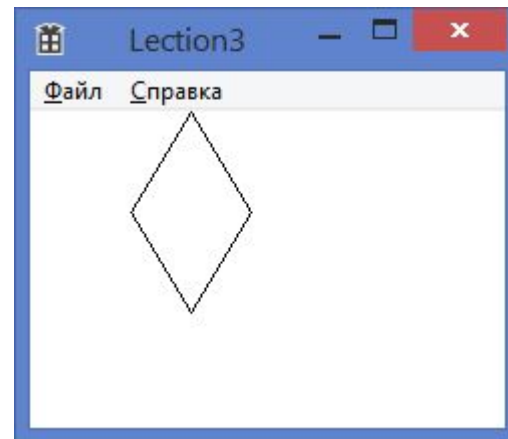
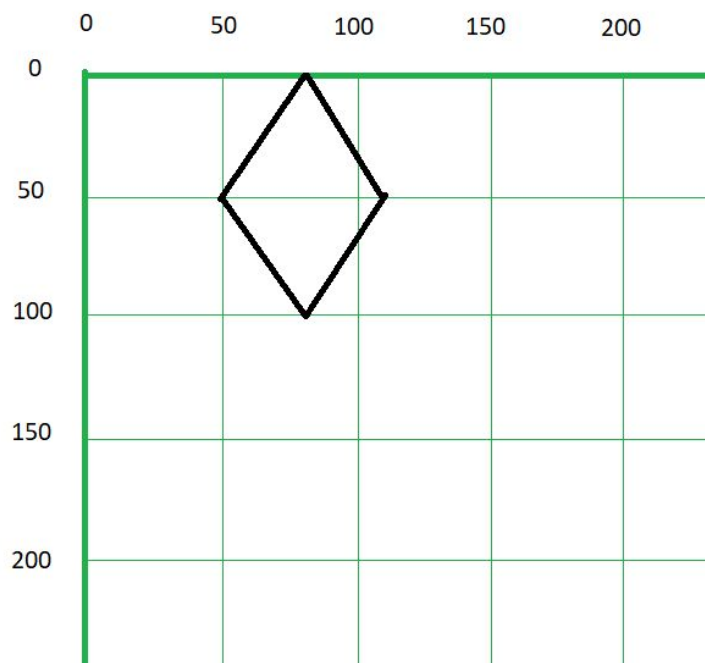
Власенко Олег Федосович

Нарисуем ромб



```
// Рисуем ромб  
MoveToEx(hdc, 80, 0, NULL);  
LineTo(hdc, 50, 50);  
LineTo(hdc, 80, 100);  
LineTo(hdc, 110, 50);  
LineTo(hdc, 80, 0);
```

Ромб



// Рисуем ромб

```
MoveToEx(hdc, 80, 0, NULL);
```

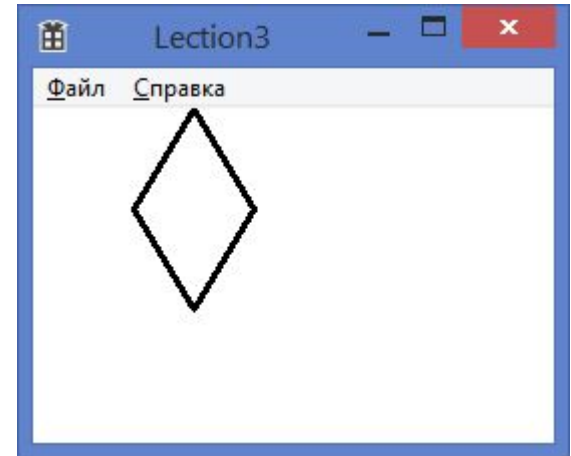
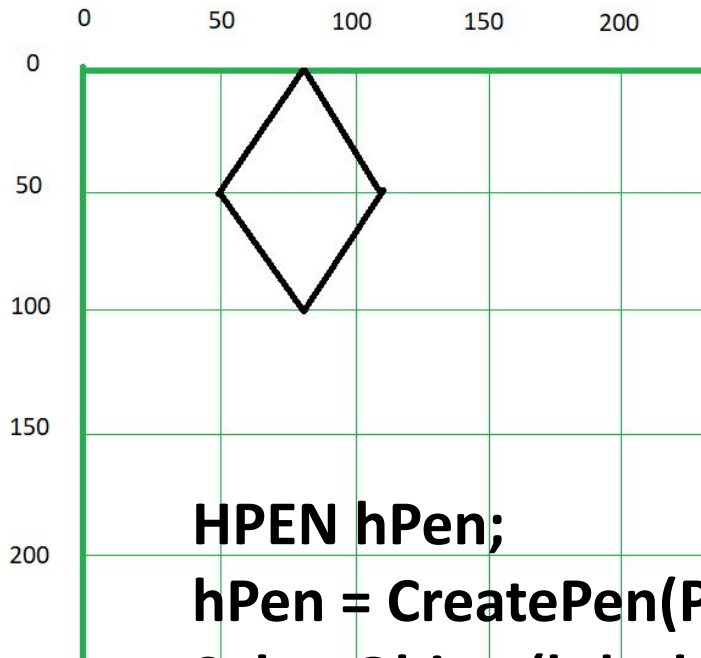
```
LineTo(hdc, 50, 50);
```

```
LineTo(hdc, 80, 100);
```

```
LineTo(hdc, 110, 50);
```

```
LineTo(hdc, 80, 0);
```

Ромб



```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 3, RGB(0, 0, 0));
```

```
SelectObject(hdc, hPen);
```

```
// Рисуем ромб
```

```
MoveToEx(hdc, 80, 0, NULL);
```

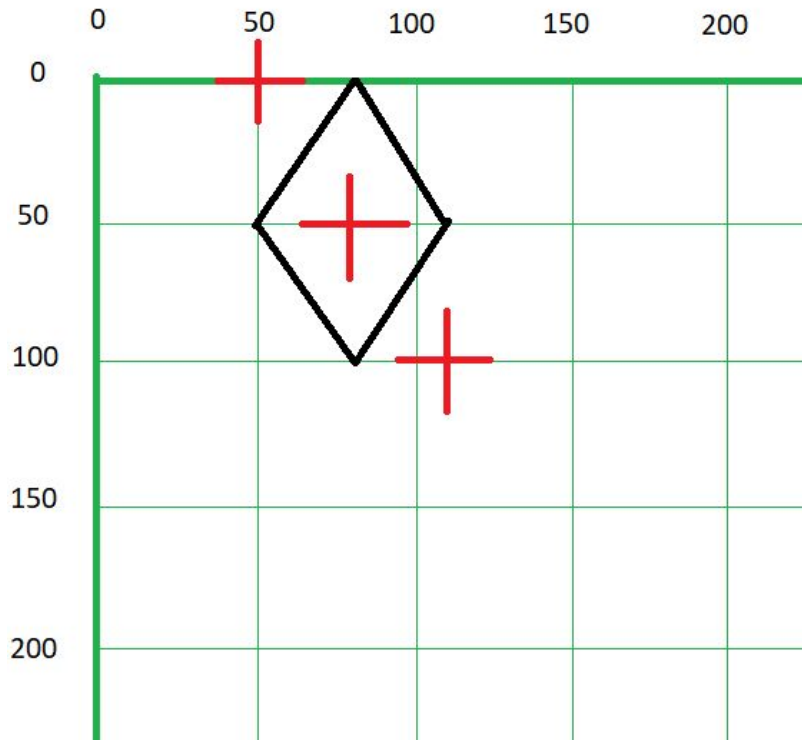
```
LineTo(hdc, 50, 50);
```

```
LineTo(hdc, 80, 100);
```

```
LineTo(hdc, 110, 50);
```

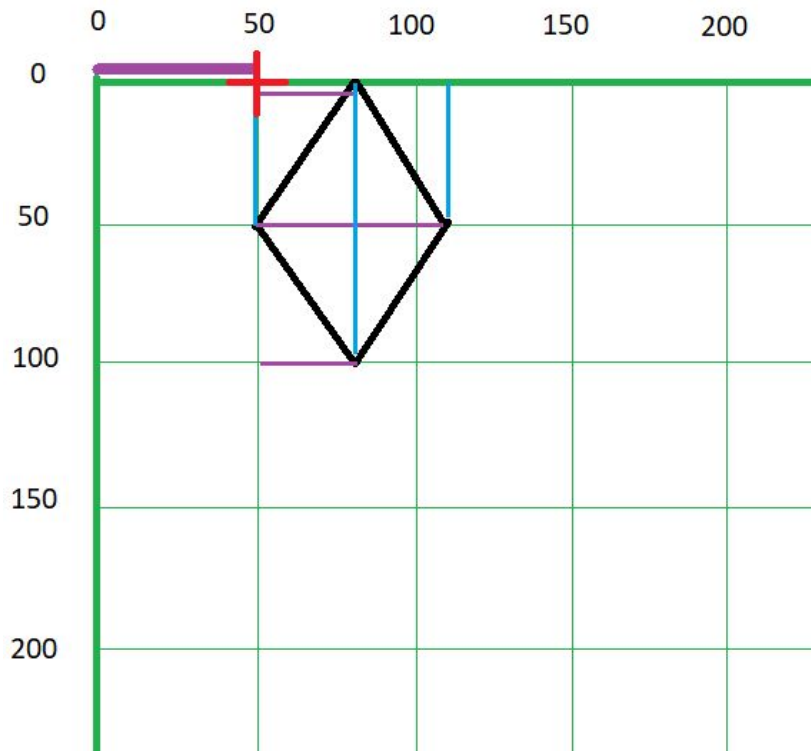
```
LineTo(hdc, 80, 0);
```

Относительные координаты



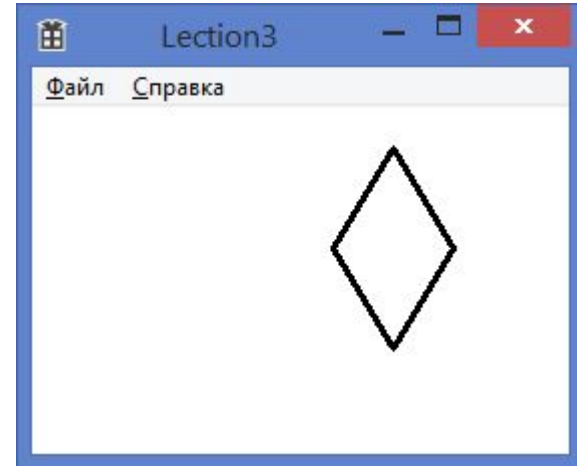
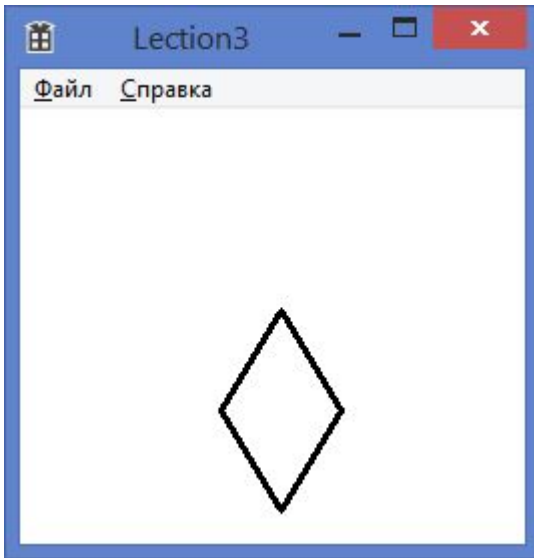
```
int x = 50;  
int y = 0;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```

Относительные координаты



```
int x = 50;  
int y = 0;  
MoveToEx(hdc, x + 30, y,  
NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```

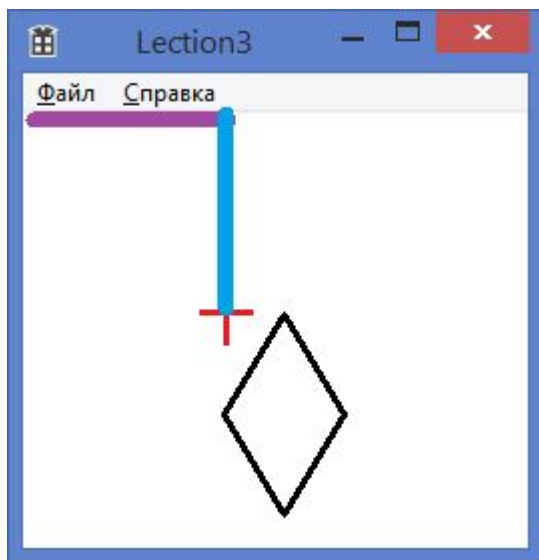
Относительные координаты



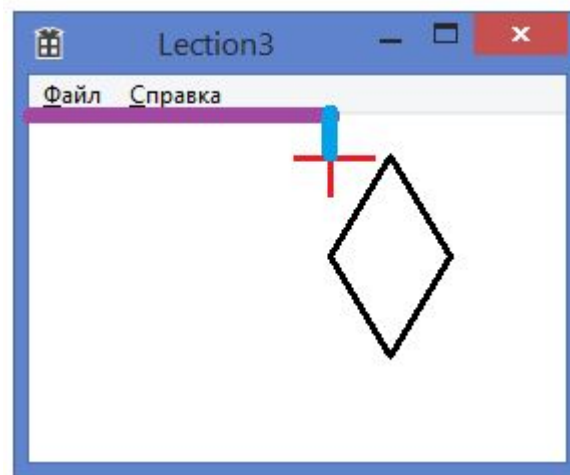
```
int x = 100;  
int y = 100;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```

```
int x = 150;  
int y = 20;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```

Относительные координаты

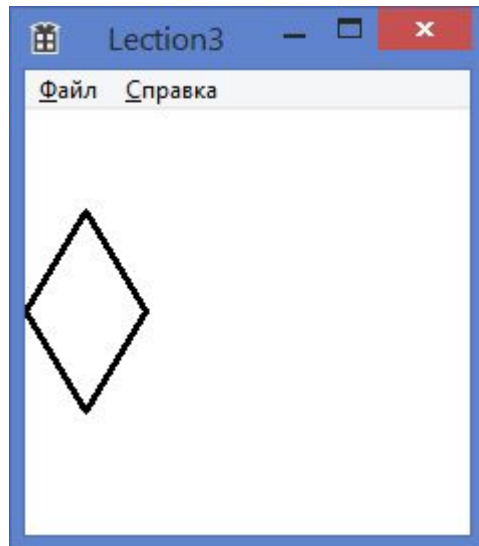
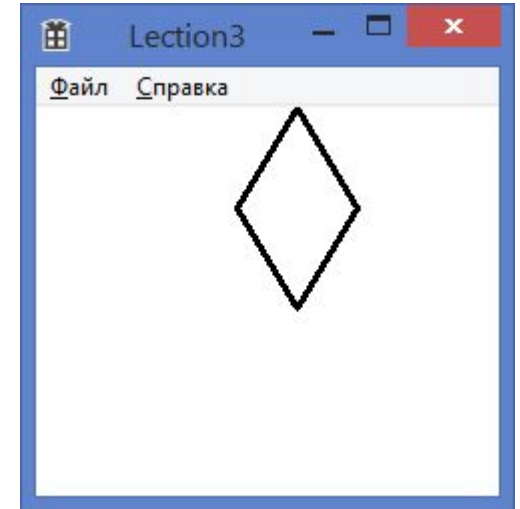
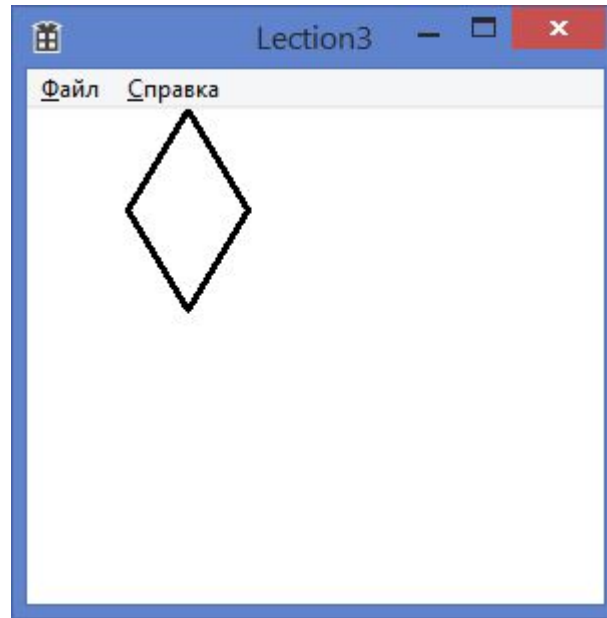
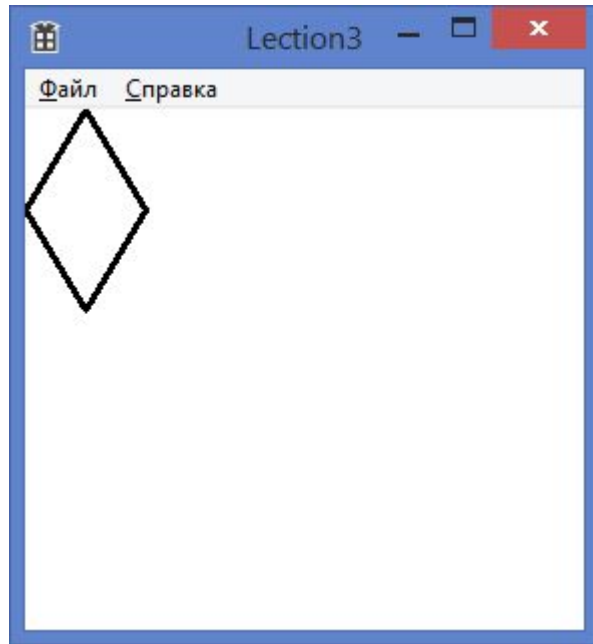


```
int x = 100;  
int y = 100;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```



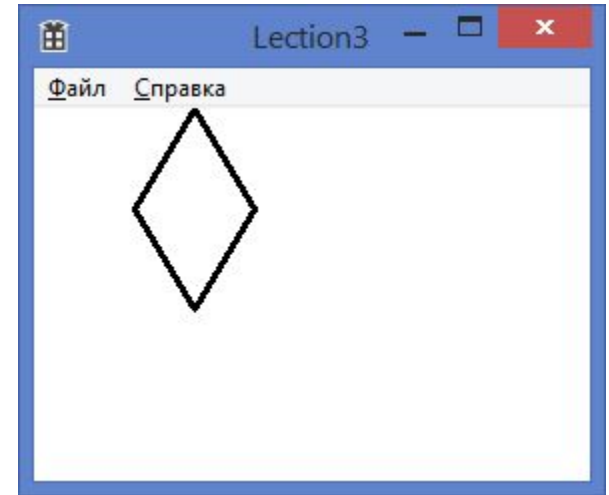
```
int x = 150;  
int y = 20;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```


Относительные координаты



Отдельная функция для отрисовки ромба с заданным положением

```
void Romb(HDC hdc, int x, int y) {  
    MoveToEx(hdc, x + 30, y, NULL);  
    LineTo(hdc, x, y + 50);  
    LineTo(hdc, x + 30, y + 100);  
    LineTo(hdc, x + 60, y + 50);  
    LineTo(hdc, x + 30, y);  
}
```



...

```
HDC hdc = BeginPaint(hWnd, &ps);  
HPEN hPen;  
hPen = CreatePen(PS_SOLID, 3, RGB(0, 0, 0));  
SelectObject(hdc, hPen);  
Romb(hdc, 50, 0);
```

...

Рисуем при помощи нашей функции несколько ромбов в ряд

...

```
HDC hdc = BeginPaint(hWnd, &ps);
```

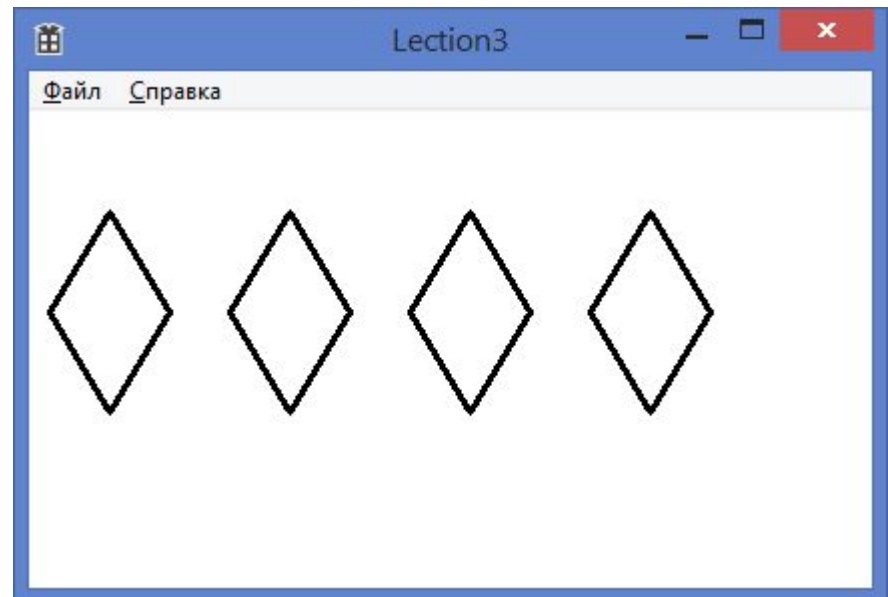
```
Romb(hdc, 10, 50);
```

```
Romb(hdc, 100, 50);
```

```
Romb(hdc, 190, 50);
```

```
Romb(hdc, 280, 50);
```

...



Рисуем при помощи нашей функции несколько ромбов – используем цикл

...

```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 3, RGB(0, 0, 0));
```

```
SelectObject(hdc, hPen);
```

```
int x = 10;
```

```
int y = 50;
```

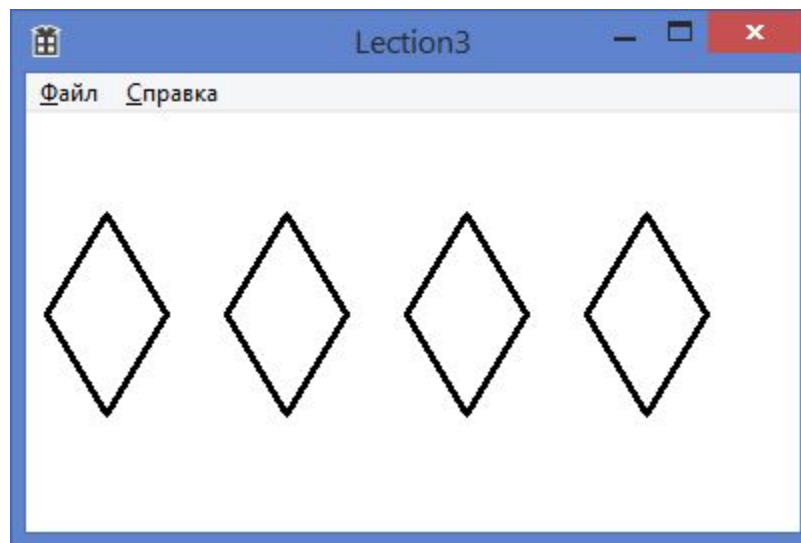
```
do {
```

```
    Romb(hdc, x, y);
```

```
    x += 90;
```

```
} while (x <= 280);
```

...



Рисуем при помощи нашей функции несколько ромбов – при помощи цикла

...

```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 3, RGB(0, 0, 0));
```

```
SelectObject(hdc, hPen);
```

```
int x = 10;
```

```
int y = 50;
```

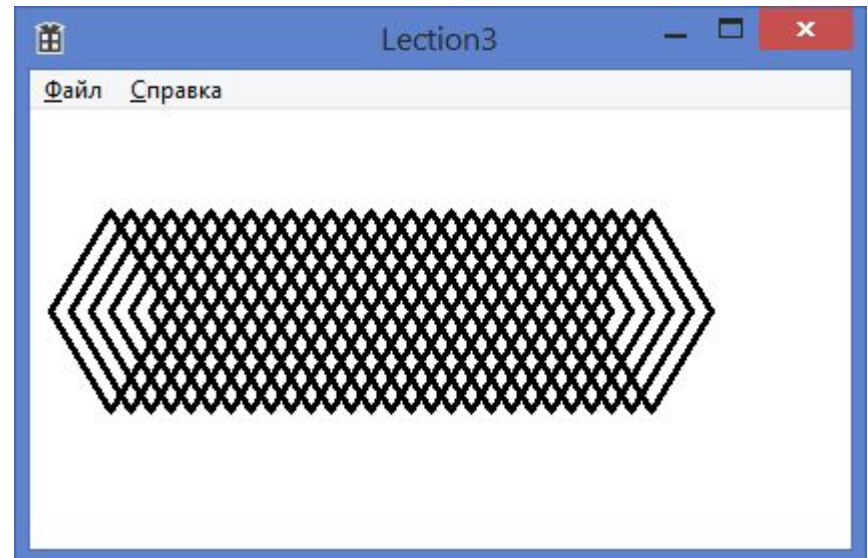
```
do {
```

```
    Romb(hdc, x, y);
```

```
    x += 10;
```

```
} while (x <= 280);
```

...



Рисуем при помощи нашей функции несколько ромбов – при помощи цикла

...

```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 1, RGB(0, 0, 0));
```

```
SelectObject(hdc, hPen);
```

```
int x = 10;
```

```
int y = 50;
```

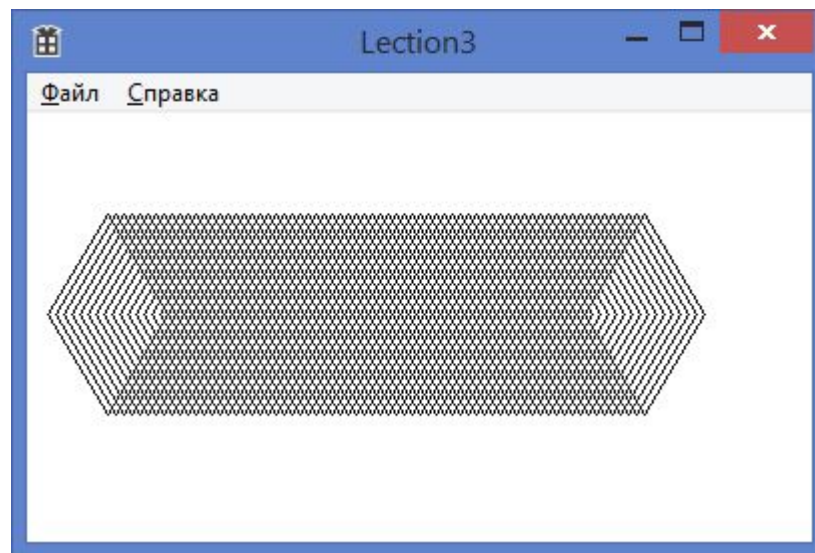
```
do {
```

```
    Romb(hdc, x, y);
```

```
    x += 4;
```

```
} while (x <= 280);
```

...



Рисуем при помощи нашей функции несколько ромбов – при помощи цикла

...

```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 1, RGB(0, 0, 0));
```

```
SelectObject(hdc, hPen);
```

```
int x = 10;
```

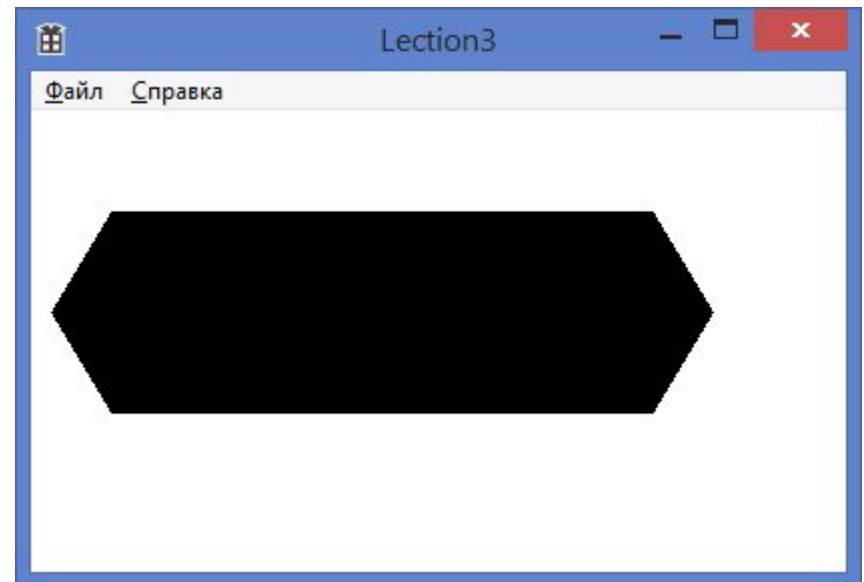
```
int y = 50;
```

```
do {
```

```
    Romb(hdc, x, y);
```

```
    x += 1;
```

```
} while (x <= 280);...
```



Трассировка циклического алгоритма

...

```
HPEN hPen;
```

```
hPen = CreatePen(PS_SOLID, 3, RGB(0, 128, 0));
```

```
SelectObject(hdc, hPen);
```

```
int x = 10;
```

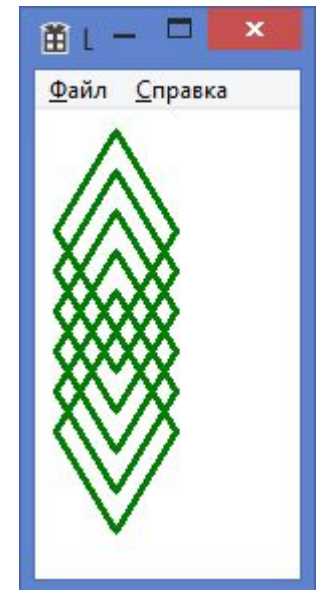
```
int y = 10;
```

```
do {
```

```
    Romb(hdc, x, y);
```

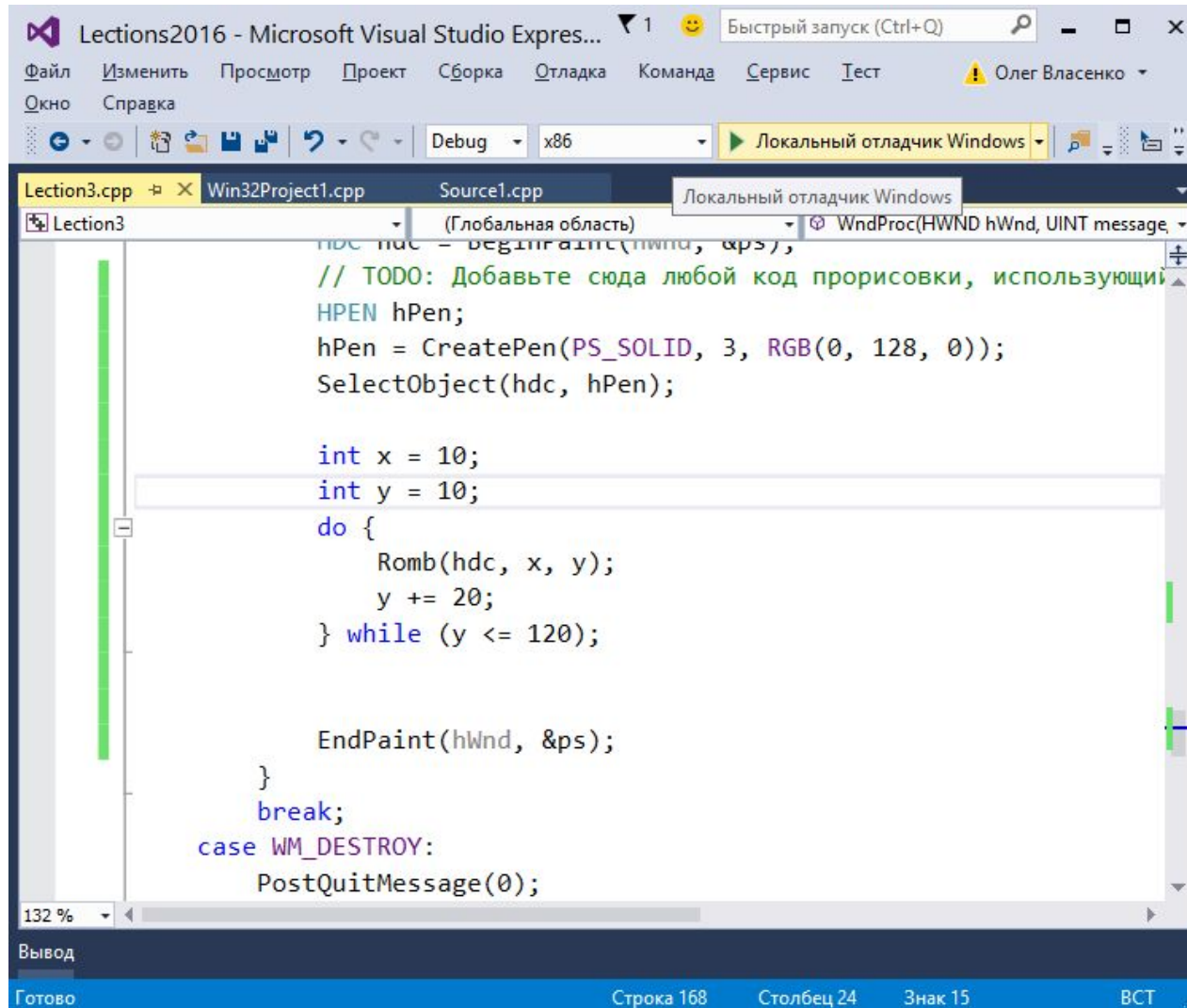
```
    y += 20;
```

```
} while (y <= 120);
```



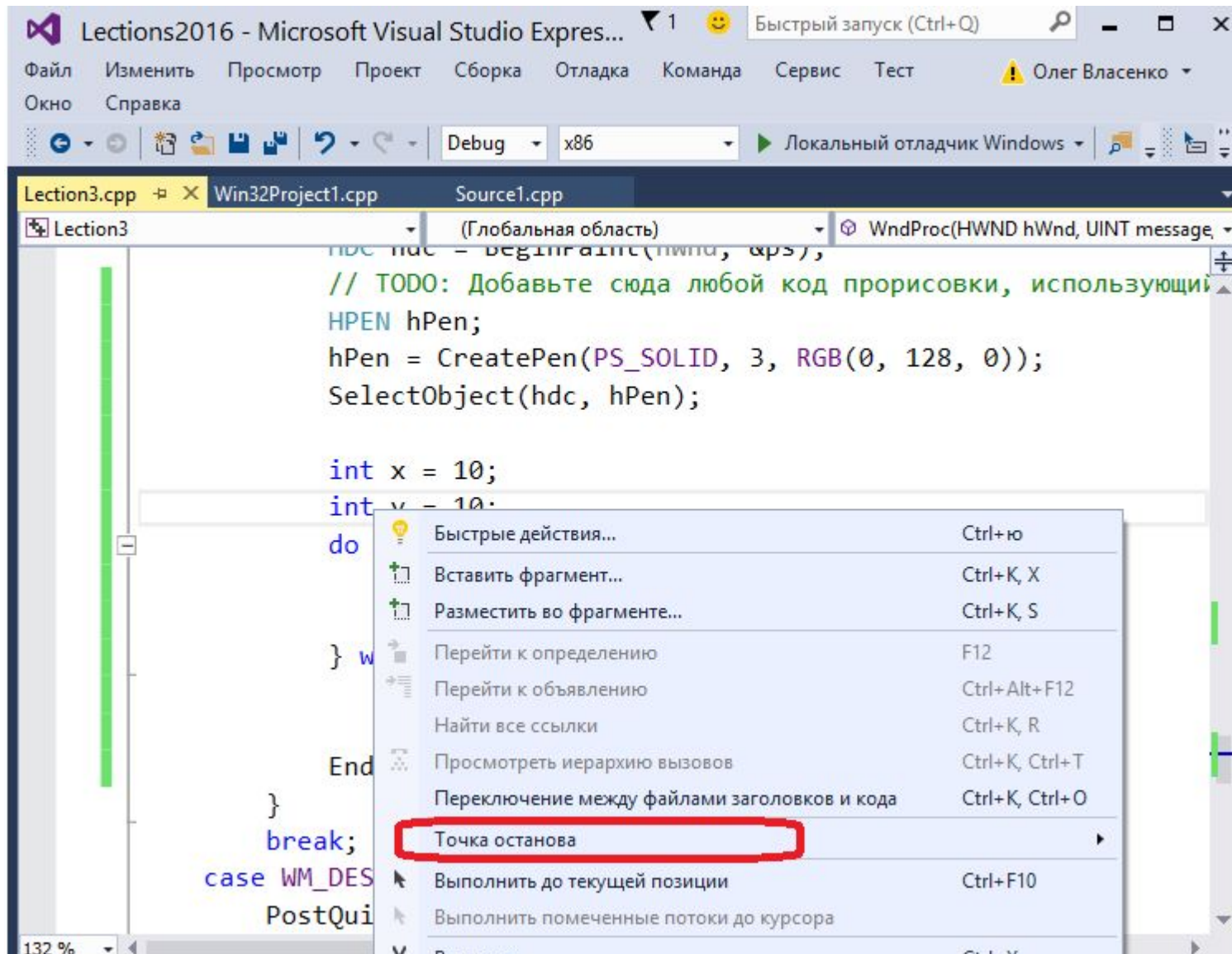
Трассировка циклического алгоритма (2)

Установка точки останова



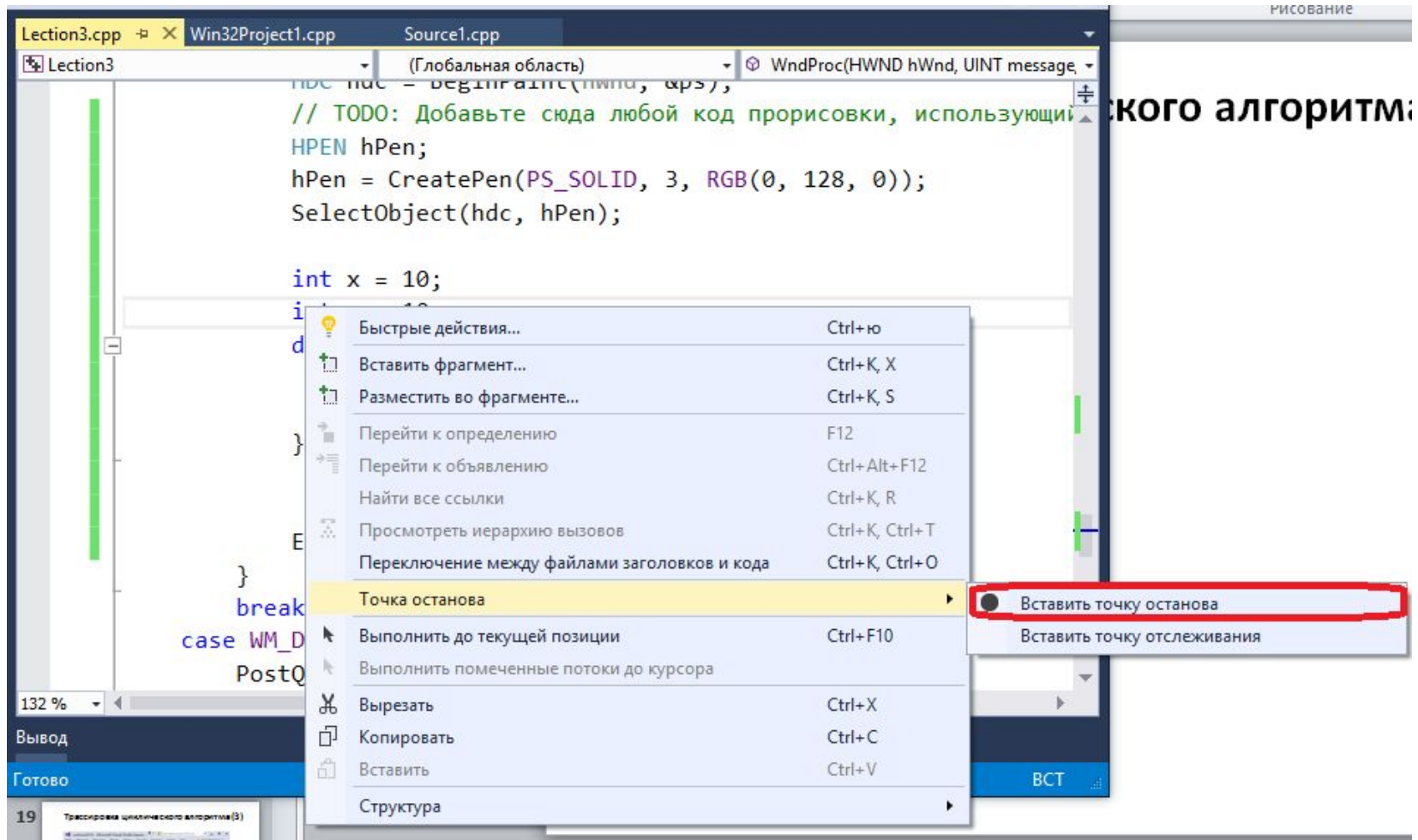
Трассировка циклического алгоритма

(3)



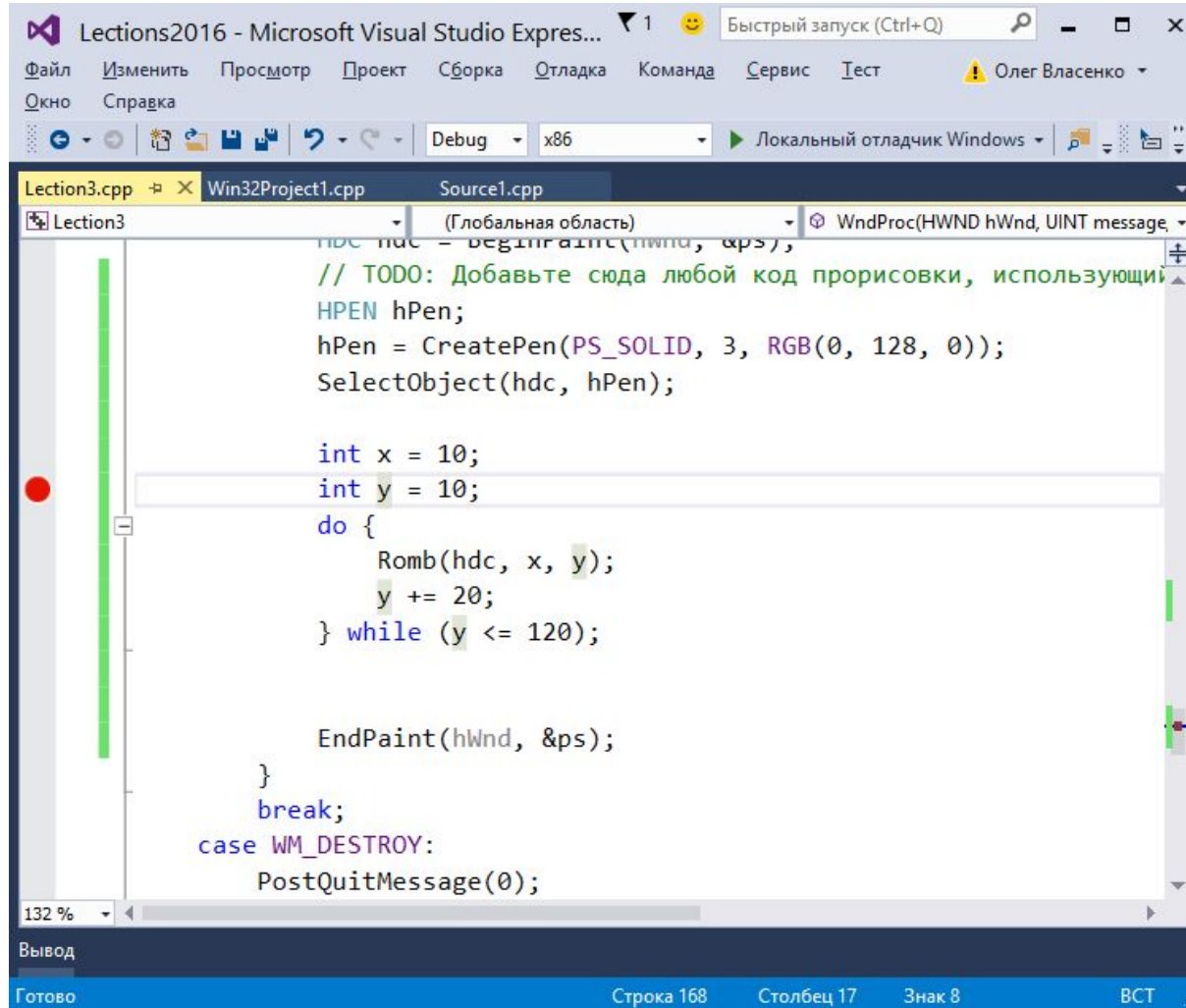
Трассировка циклического алгоритма

(4)



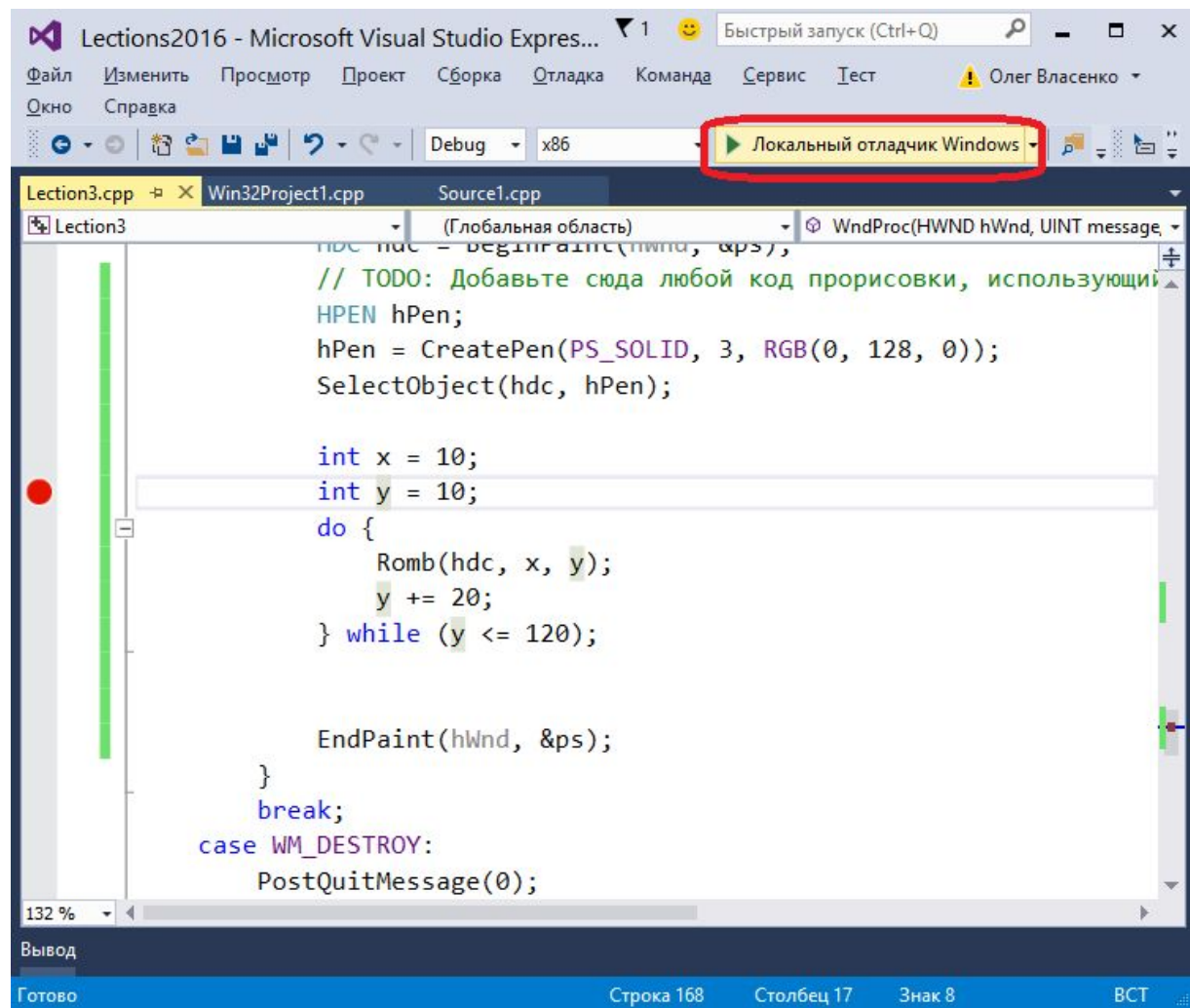
Трассировка циклического алгоритма

(5)



Трассировка циклического алгоритма (6)

Запуск трассировки



Трассировка циклического алгоритма (7)

(при)остановка в точке останова

Microsoft Visual Studio - Lessons2016 (Отладка)

Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест

Окно Справка

Процесс: [7264] Lesson3.exe

События жизненного цикла Поток: [9992] Основной поток

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
hdc = BeginPaint(hWnd, &ps);  
// TODO: Добавьте сюда любой код прорисовки, использующий  
HPEN hPen;  
hPen = CreatePen(PS_SOLID, 3, RGB(0, 128, 0));  
SelectObject(hdc, hPen);  
  
int x = 10;  
int y = 10;  
do {  
    Romb(hdc, x, y);  
}
```

132 %

Локальные

Имя	Значение	Тип
y	-858993460	int
hPen	0xa930090a {unused=???}	HPEN__
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAINT	PAINTSTRUCT
hdc	0x4b0127ff {unused=???}	HDC__
hWnd	0x00150a46 {unused=???}	HWND__
message	15	unsigned int
wParam	0	unsigned int
lParam	0	long

Локальные Контрольные значения 1

Стек вызовов

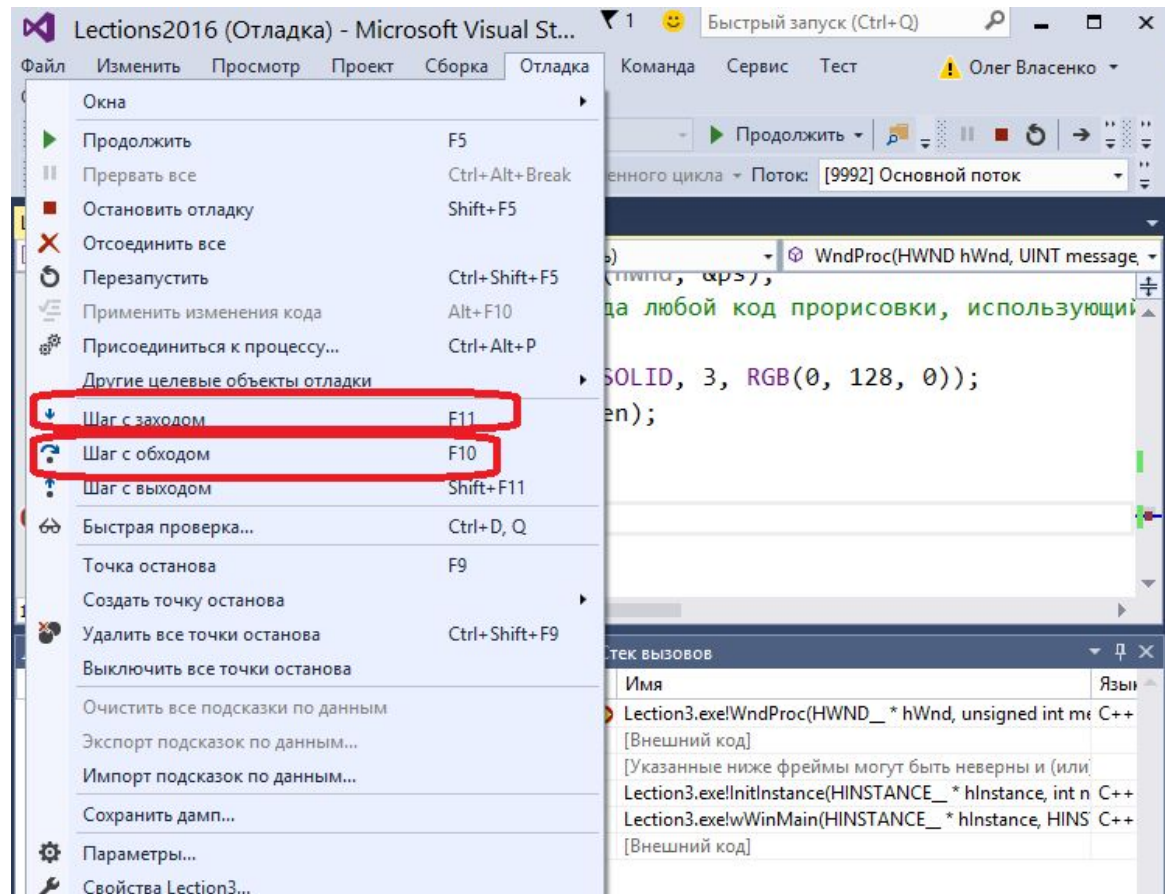
Имя	Язык
Lesson3.exe!WndProc(HWND__ * hWnd, unsigned int message)	C++
[Внешний код]	
Lesson3.exe!InitInstance(HINSTANCE__ * hInstance, int nCmdShow)	C++
Lesson3.exe!WinMain(HINSTANCE__ * hInstance, HINSTANCE__ * hPrevInstance, LPSTR lpCmdLine, int nCmdShow)	C++
[Внешний код]	

Загрузка символов для oleacc.dll

Строка 168 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (8)

Пошаговая трассировка



Трассировка циклического алгоритма (9)

Шаг сделан (F10)

Microsoft Visual Studio - Lesson3.exe (Отладка)

Файл | Изменить | Просмотр | Проект | Сборка | Отладка | Команда | Сервис | Тест | Олер Власенко

Окно | Справка

Процесс: [7264] Lesson3.exe | События жизненного цикла | Поток: [9992] Основной поток

Source1.cpp | Lesson3 | (Глобальная область) | WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y); ≤ 1 мс прошло
    y += 20;
} while (y <= 120);

EndPaint(hWnd, &ps);
```

132 %

Имя	Значение	Тип
y	10	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAINT	tagPAINT
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_
message	15	unsigned
wParam	0	unsigned
lParam	0	long

Локальные | Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int m...	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int n...	C++
Lesson3.exe!WinMain(HINSTANCE_ * hInstance, HINS...	C++
[Внешний код]	

Загрузка символов для oleacc.dll | Строка 170 | Столбец 1 | Знак 1 | ВСТ

Трассировка циклического алгоритма (10)

Еще шаг сделан (F10)

Visual Studio interface showing the debugger for Lesson3.exe. The code is a do-while loop for drawing a rectangle. The debugger is paused at the line `y += 20;`. The 'Locals' window shows variables `x`, `y`, `hPen`, `ps`, `hdc`, `hWnd`, `message`, and `wParam`. The 'Call Stack' window shows the current function and its callers.

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y);
    y += 20; ≤ 1 мс прошло
} while (y <= 120);
```

Имя	Значение	Тип
x	10	int
y	10	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra	tagPAIN
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_*
message	15	unsigned
wParam	0	unsigned

Имя	Язык
Lesson3.exe!WndProc(HWND hWnd, unsigned int message, WPARAM wParam, LPARAM lParam)	C++
[Внешний код]	
Lesson3.exe!InitInstance(HINSTANCE hInstance, int nCmdShow)	C++
Lesson3.exe!WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nCmdShow)	C++
[Внешний код]	

Локальные: Контрольные значения 1

Загрузка символов для oleacc.dll

Строка 171 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (11)

Еще шаг сделан (F10)

Lections2016 (Отладка) - Microsoft Visual St... Быстрый запуск (Ctrl+Q)

Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест ! Олег Власенко

Окно Справка

Debug x86 Продолжить

Процесс: [7264] Lesson3.exe События жизненного цикла Поток: [9992] Основной поток

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y);
    y += 20;
} while (y <= 120);
```

132 %

Локальные

Имя	Значение	Тип
x	10	int
y	30	int
hPen	0xa930090a {unused=??}	HPEN_
ps	{hdc=0x4b0127ff {unused=??} fEra tagPAIN	
hdc	0x4b0127ff {unused=??}	HDC_*
hWnd	0x00150a46 {unused=??}	HWND_
message	15	unsigned
wParam	0	unsigned

Локальные Контрольные значения 1

Стек вызовов

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int m...	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int n...	C++
Lesson3.exe!WinMain(HINSTANCE_ * hInstance, HINS...	C++
[Внешний код]	

Загрузка символов для oleacc.dll

Строка 172 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (12)

Еще шаг сделан

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y); ≤ 1 мс прошло
    y += 20;
} while (y <= 120);
```

132 %

Имя	Значение	Тип
x	10	int
y	30	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAIN	tagPAIN
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_*
message	15	unsigned
wParam	0	unsigned

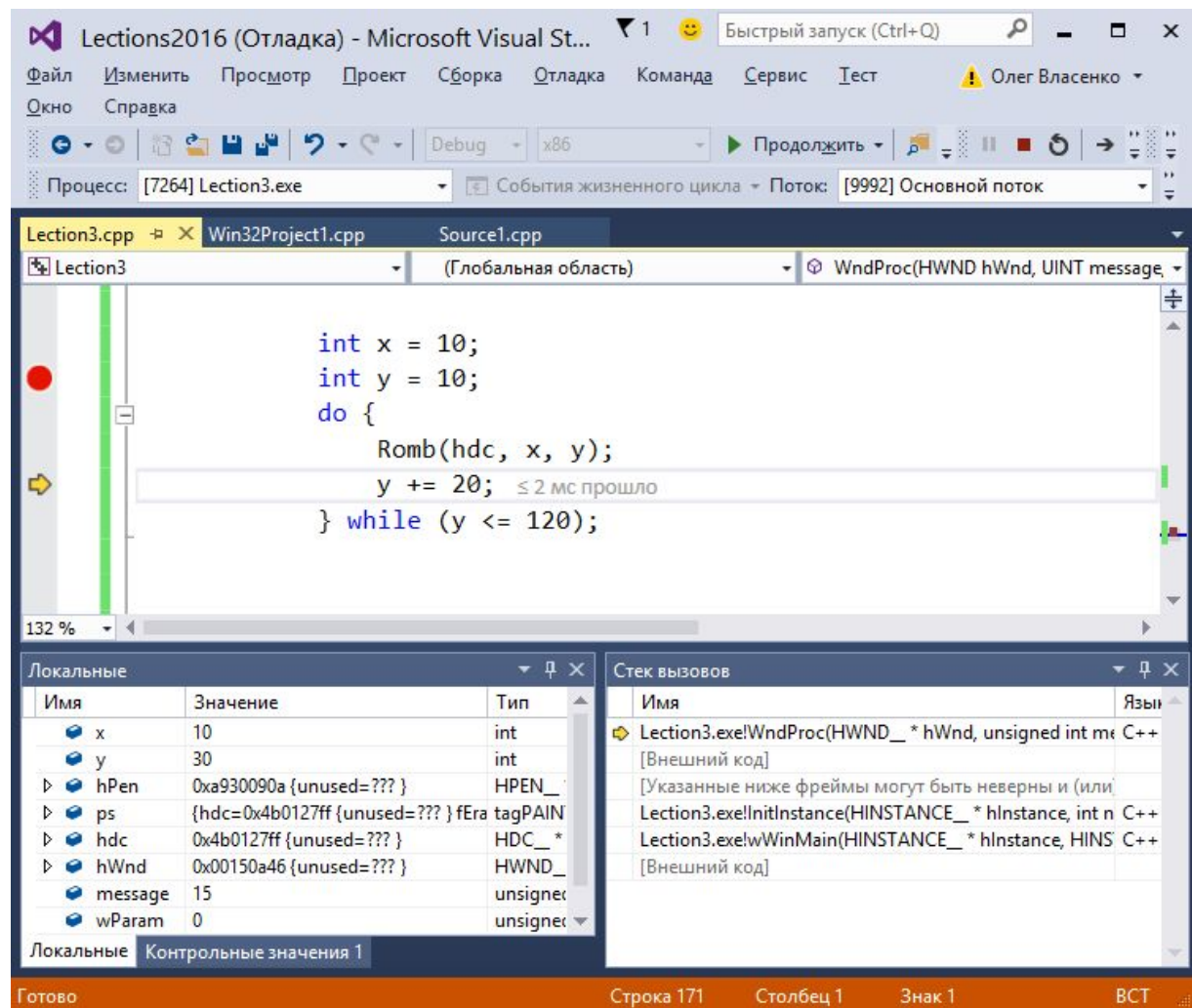
Локальные Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int m...	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int n	C++
Lesson3.exe!wWinMain(HINSTANCE_ * hInstance, HINS	C++
[Внешний код]	

Готово Строка 170 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (13)

Еще шаг сделан



Lections2016 (Отладка) - Microsoft Visual St... 1 Быстрый запуск (Ctrl+Q)

Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест Олер Власенко

Окно Справка

Процесс: [7264] Lesson3.exe События жизненного цикла Поток: [9992] Основной поток

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y);
    y += 20; ≤ 2 мс прошло
} while (y <= 120);
```

132 %

Локальные

Имя	Значение	Тип
x	10	int
y	30	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAIN	
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_
message	15	unsigned
wParam	0	unsigned

Локальные Контрольные значения 1

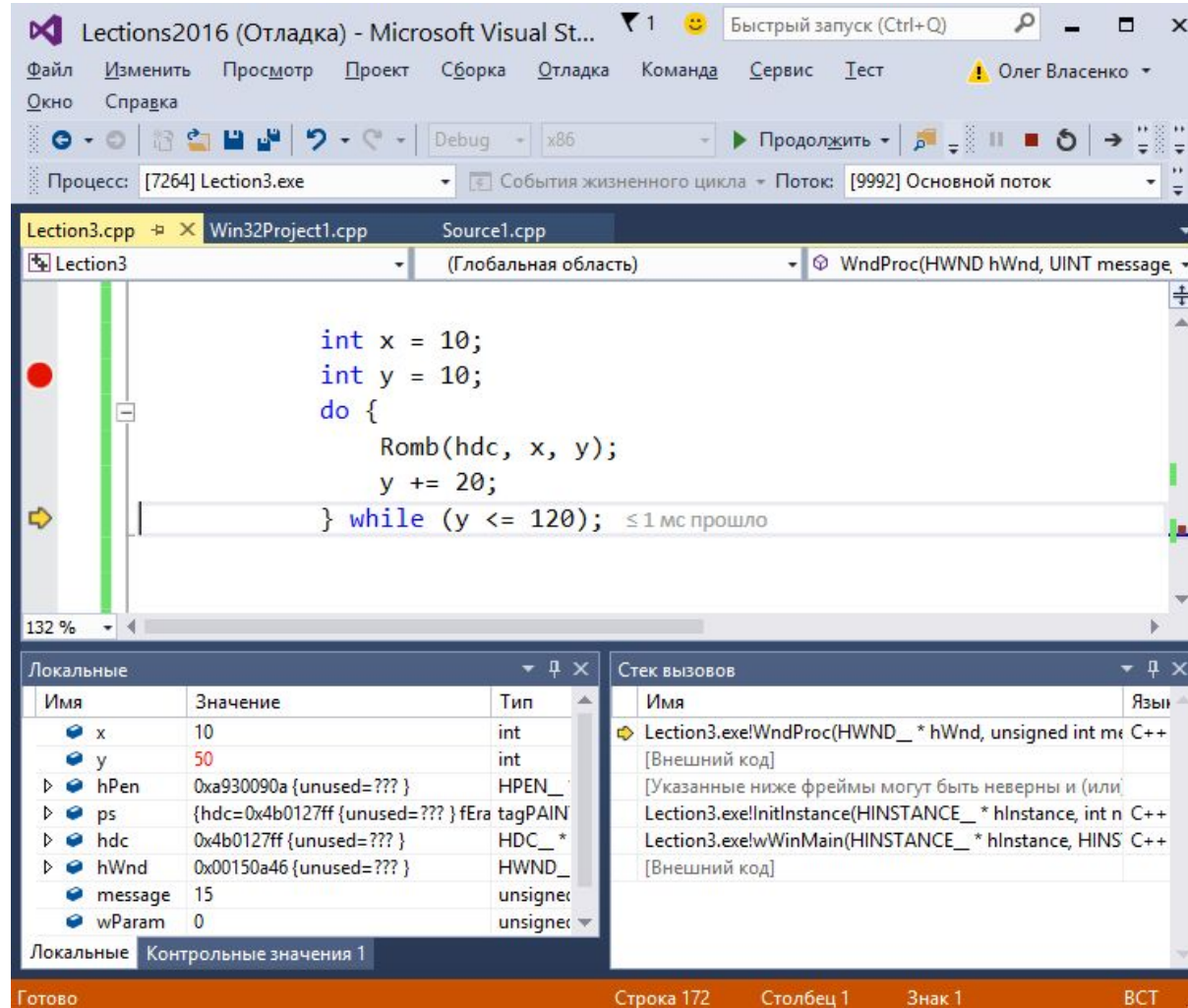
Стек вызовов

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int me	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int n	C++
Lesson3.exe!wWinMain(HINSTANCE_ * hInstance, HINS	C++
[Внешний код]	

Готово Строка 171 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (14)

Еще шаг сделан



Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y);
    y += 20;
} while (y <= 120); ≤ 1 мс прошло
```

132 %

Имя	Значение	Тип
x	10	int
y	50	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAINT	
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_
message	15	unsigned
wParam	0	unsigned

Локальные Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int message)	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или) неактуальны]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int nCmdShow)	C++
Lesson3.exe!wWinMain(HINSTANCE_ * hInstance, HINSTANCE_ * hPrevInstance, LPWSTR lpCmdLine, int nCmdShow)	C++
[Внешний код]	

Готово Строка 172 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (15)

Еще шаг сделан

Microsoft Visual Studio (Отладка) - Lessons2016 (Отладка)

Быстрый запуск (Ctrl+Q)

Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест Олер Власенко

Окно Справка

Процесс: [7264] Lesson3.exe События жизненного цикла Поток: [9992] Основной поток

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y); ≤ 1 мс прошло
    y += 20;
} while (y <= 120);
```

132 %

Локальные

Имя	Значение	Тип
x	10	int
y	50	int
hPen	0xa930090a {unused=???}	HPEN_
ps	{hdc=0x4b0127ff {unused=???} fEra tagPAINT	
hdc	0x4b0127ff {unused=???}	HDC_ *
hWnd	0x00150a46 {unused=???}	HWND_
message	15	unsigned
wParam	0	unsigned

Локальные Контрольные значения 1

Стек вызовов

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int message)	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или)]	
Lesson3.exe!InitInstance(HINSTANCE_ * hInstance, int nCmdShow)	C++
Lesson3.exe!WinMain(HINSTANCE_ * hInstance, HINSTANCE_ * hPrevInstance, LPSTR lpszCmdLine, int nCmdShow)	C++
[Внешний код]	

Готово Строка 170 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (16)

Возобновить выполнение

The screenshot shows the Microsoft Visual Studio 2016 interface during a debugging session. The title bar reads 'Lectons2016 (Отладка) - Microsoft Visual St...'. The menu bar includes 'Файл', 'Изменить', 'Просмотр', 'Проект', 'Сборка', 'Отладка', 'Команда', 'Сервис', and 'Тест'. The toolbar shows various debugging icons, with the 'Continue' button (a green play icon) highlighted by a red rectangle. Below the toolbar, the 'Процесс' dropdown shows '[7264] Lesson3.exe'. The main editor window displays the source code for 'Lesson3.cpp' at the 'Global scope' level, showing a function 'WndProc(HWND hWnd, UINT message)'. The code includes variable declarations for 'x' and 'y', and a 'do' loop that calls 'Romb(hdc, x, y);' and increments 'y' until it reaches 120. A comment '≤ 1 мс прошло' is present next to the 'Romb' call. The 'Locals' window at the bottom left shows the current state of variables: 'x' is 10, 'y' is 50, 'hPen' is 0xa930090a, 'ps' is {hdc=0x4b0127ff, fEra=tagPAIN}, 'hdc' is 0x4b0127ff, 'hWnd' is 0x00150a46, 'message' is 15, and 'wParam' is 0. The 'Call Stack' window at the bottom right shows the current call stack, with the top entry being 'Lesson3.exe!WndProc(HWND__ * hWnd, unsigned int message)' in C++.

Лекции2016 (Отладка) - Microsoft Visual St... Быстрый запуск (Ctrl+Q)

Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест Олег Власенко

Окно Справка

Debug x86 Продолжить

Процесс: [7264] Lesson3.exe События жизненного цикла Поток: [5992] Основной поток

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message)

```
int x = 10;
int y = 10;
do {
    Romb(hdc, x, y); ≤ 1 мс прошло
    y += 20;
} while (y <= 120);
```

132 %

Имя	Значение	Тип
x	10	int
y	50	int
hPen	0xa930090a {unused=???}	HPEN
ps	{hdc=0x4b0127ff {unused=???} fEra=tagPAIN}	tagPAINT
hdc	0x4b0127ff {unused=???}	HDC
hWnd	0x00150a46 {unused=???}	HWND
message	15	unsigned int
wParam	0	unsigned int

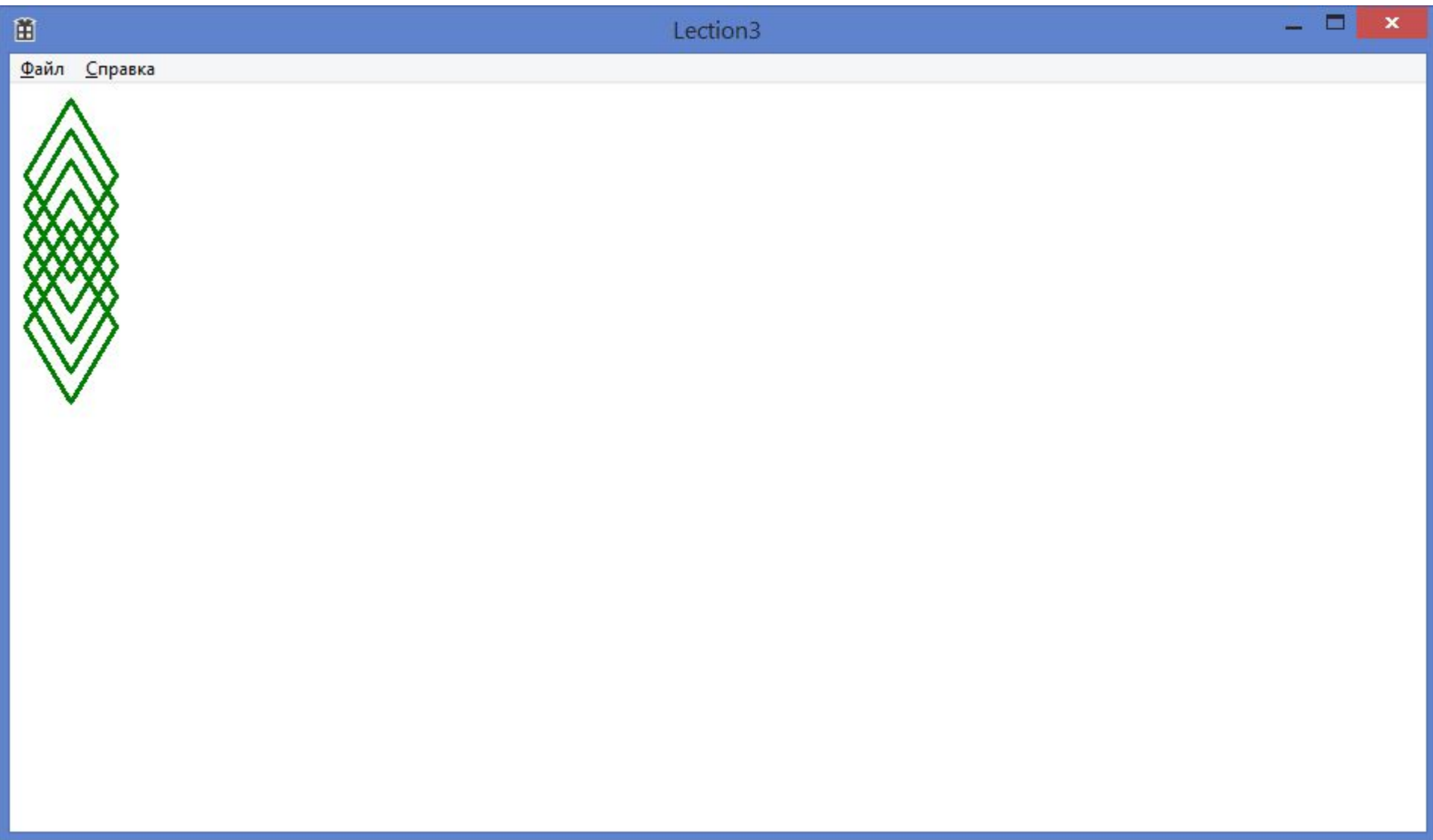
Локальные Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND__ * hWnd, unsigned int message)	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или) неактуальны]	
Lesson3.exe!InitInstance(HINSTANCE__ * hInstance, int nCmdShow)	C++
Lesson3.exe!wWinMain(HINSTANCE__ * hInstance, HINSTANCE__ * hPrevInstance, LPWSTR lpCmdLine, int nCmdShow)	C++
[Внешний код]	

Готово Строка 170 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (17)

Приложение работает!



Изменяем сразу и X и Y

...

```
int x = 10;
```

```
int y = 50;
```

```
do {
```

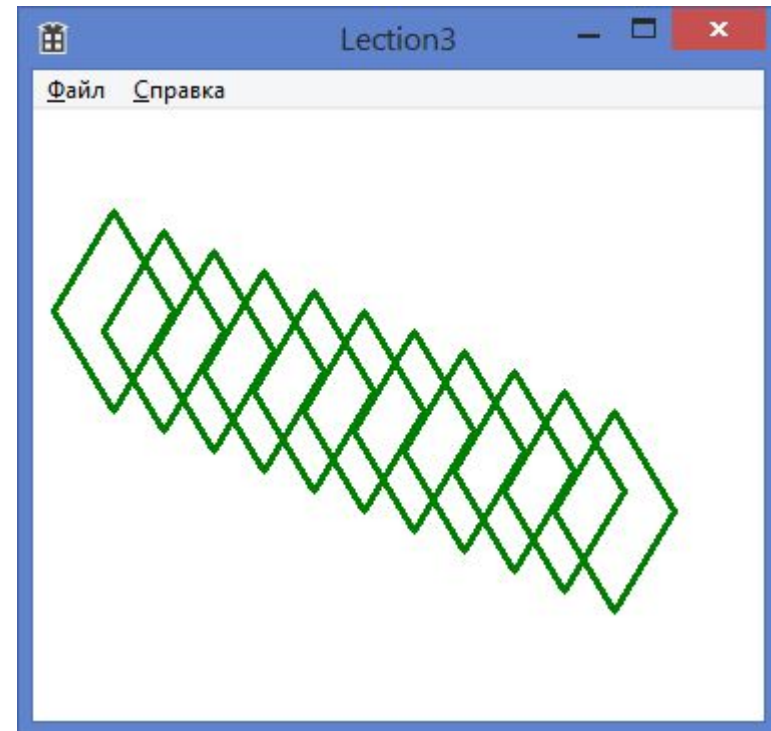
```
    Romb(hdc, x, y);
```

```
    x += 25;
```

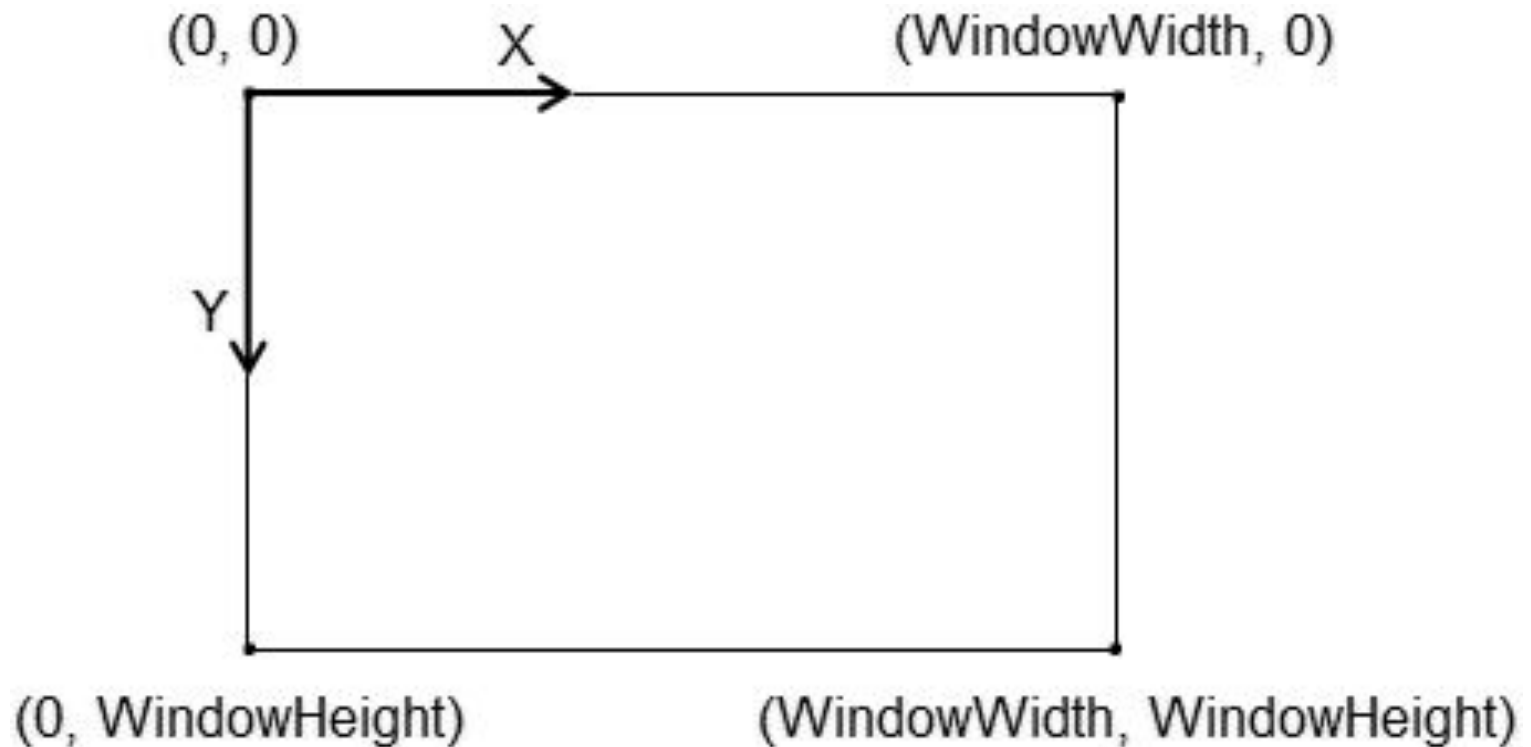
```
    y += 10;
```

```
} while (x <= 280);
```

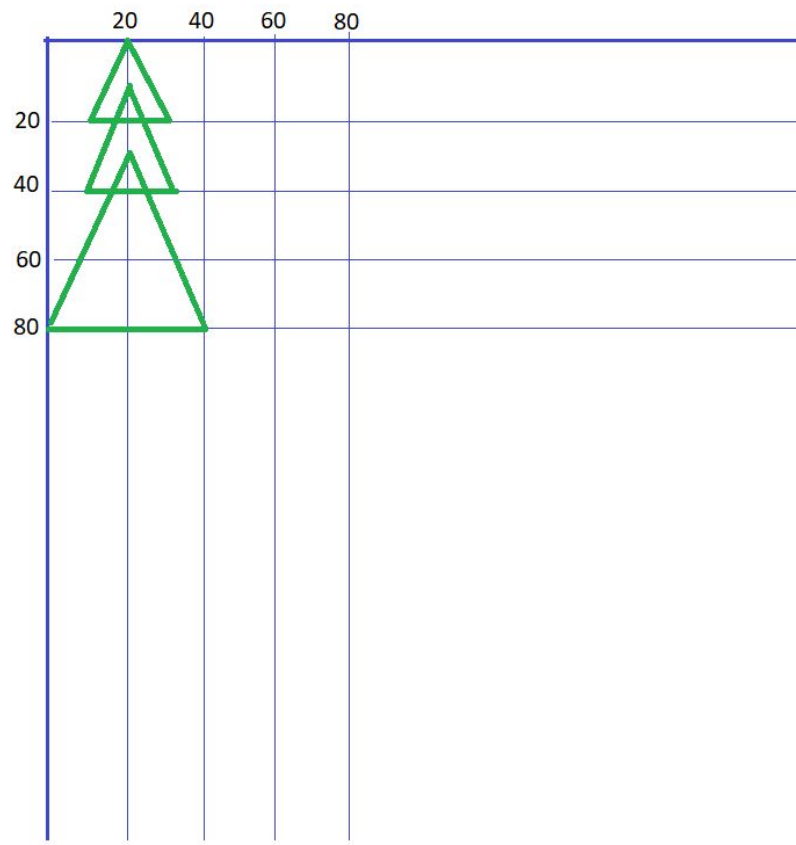
...



Автомобиль деда Мороза - логотип



Автомобиль деда Мороза - логотип



Автомобиль деда Мороза - логотип

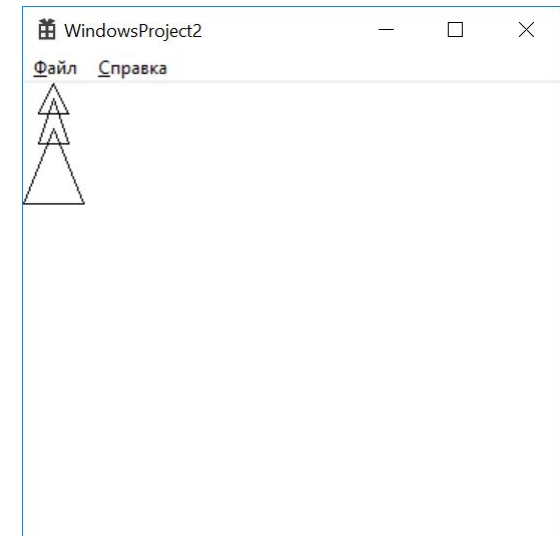
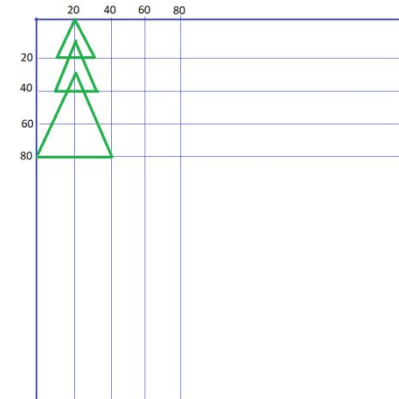
```
case WM_PAINT:
{
    PAINTSTRUCT ps;
    HDC hdc = BeginPaint(hWnd, &ps);
    // TODO: Добавьте сюда любой код прорисовки, использующий HDC...

    // верхний треугольник
    MoveToEx(hdc, 20, 0, NULL);
    LineTo(hdc, 30, 20);
    LineTo(hdc, 10, 20);
    LineTo(hdc, 20, 0);

    // средний треугольник
    MoveToEx(hdc, 20, 10, NULL);
    LineTo(hdc, 30, 40);
    LineTo(hdc, 10, 40);
    LineTo(hdc, 20, 10);

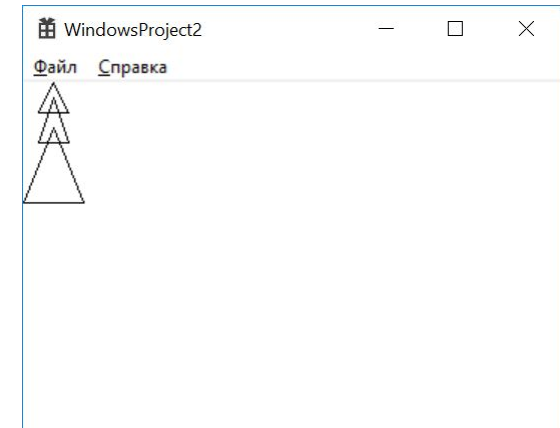
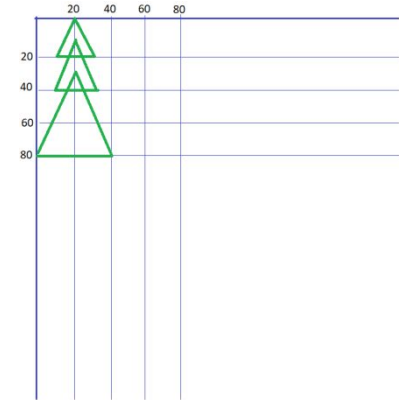
    // нижний треугольник
    MoveToEx(hdc, 20, 30, NULL);
    LineTo(hdc, 40, 80);
    LineTo(hdc, 0, 80);
    LineTo(hdc, 20, 30);

    EndPaint(hWnd, &ps);
}
break;
```



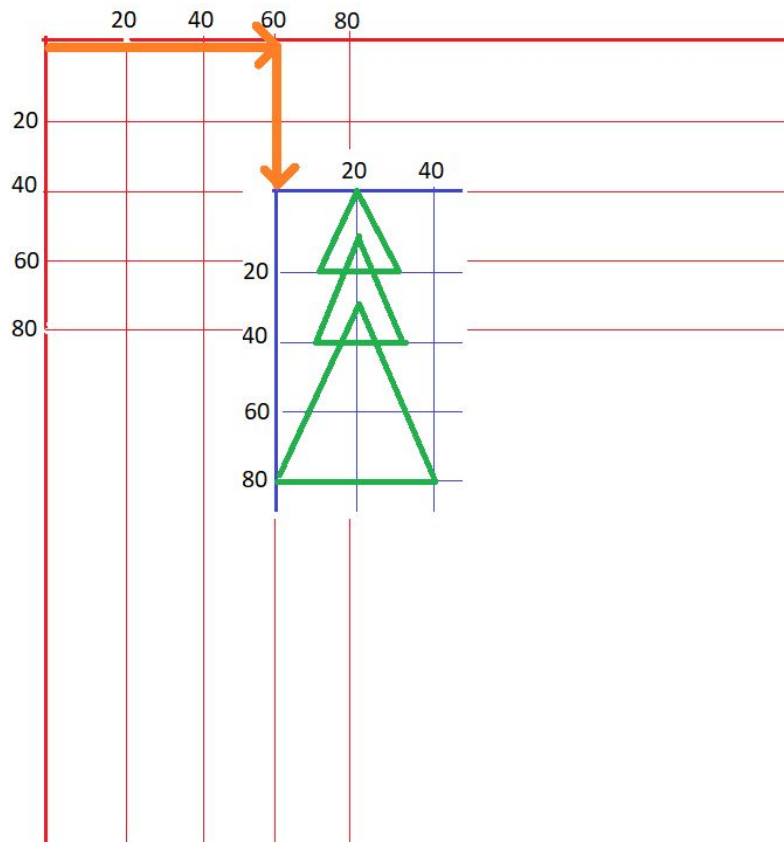
Вынесли код в отдельную функцию

```
void StClausAuto(HDC hdc) {  
    // верхний треугольник  
    MoveToEx(hdc, 20, 0, NULL);  
    LineTo(hdc, 30, 20);  
    LineTo(hdc, 10, 20);  
    LineTo(hdc, 20, 0);  
  
    // средний треугольник  
    MoveToEx(hdc, 20, 10, NULL);  
    LineTo(hdc, 30, 40);  
    LineTo(hdc, 10, 40);  
    LineTo(hdc, 20, 10);  
  
    // нижний треугольник  
    MoveToEx(hdc, 20, 30, NULL);  
    LineTo(hdc, 40, 80);  
    LineTo(hdc, 0, 80);  
    LineTo(hdc, 20, 30);  
}  
...
```



```
case WM_PAINT:  
{  
    PAINTSTRUCT ps;  
    HDC hdc = BeginPaint(hWnd, &ps);  
    // TODO: Добавьте сюда любой код прорисовки, использующий HDC...  
  
    StClausAuto(hdc);  
  
    EndPaint(hWnd, &ps);  
}  
break;
```

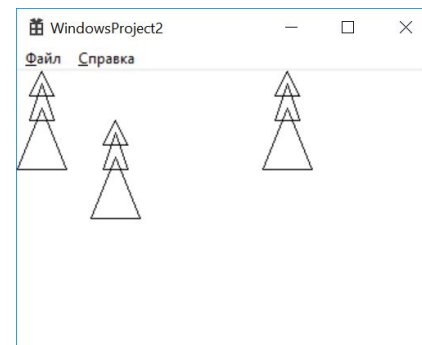
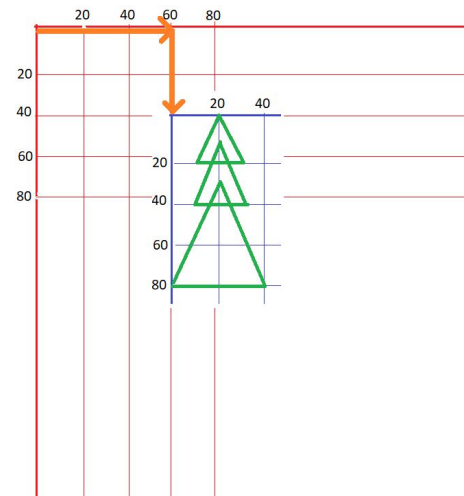
Относительные координаты



Относительные координаты

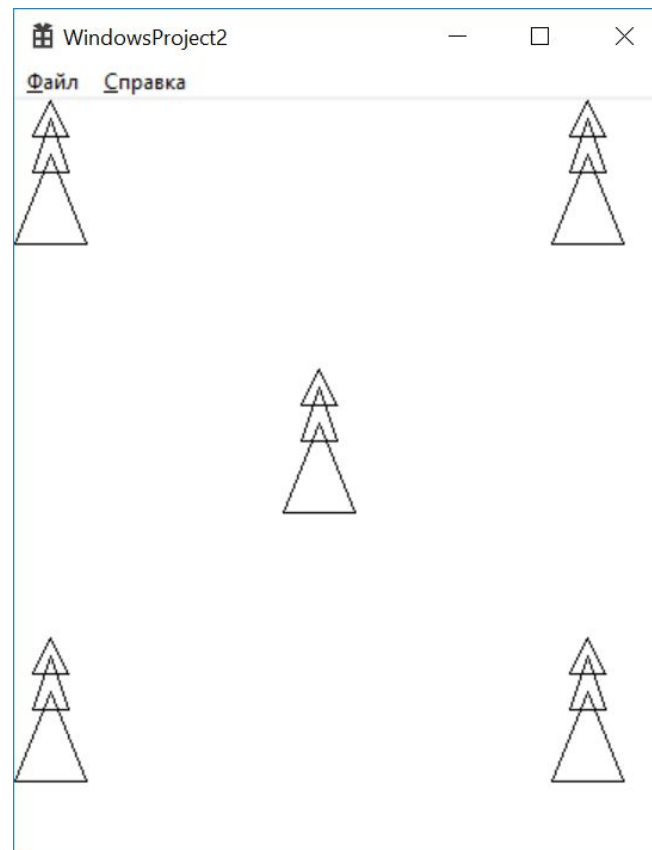
```
void StClausAuto(HDC hdc, int x, int y) {  
    // верхний треугольник  
    MoveToEx(hdc, 20 + x, 0 + y, NULL);  
    LineTo(hdc, 30 + x, 20 + y);  
    LineTo(hdc, 10 + x, 20 + y);  
    LineTo(hdc, 20 + x, 0 + y);  
  
    // средний треугольник  
    MoveToEx(hdc, 20 + x, 10 + y, NULL);  
    LineTo(hdc, 30 + x, 40 + y);  
    LineTo(hdc, 10 + x, 40 + y);  
    LineTo(hdc, 20 + x, 10 + y);  
  
    // нижний треугольник  
    MoveToEx(hdc, 20 + x, 30 + y, NULL);  
    LineTo(hdc, 40 + x, 80 + y);  
    LineTo(hdc, 0 + x, 80 + y);  
    LineTo(hdc, 20 + x, 30 + y);  
}
```

```
...  
}
```



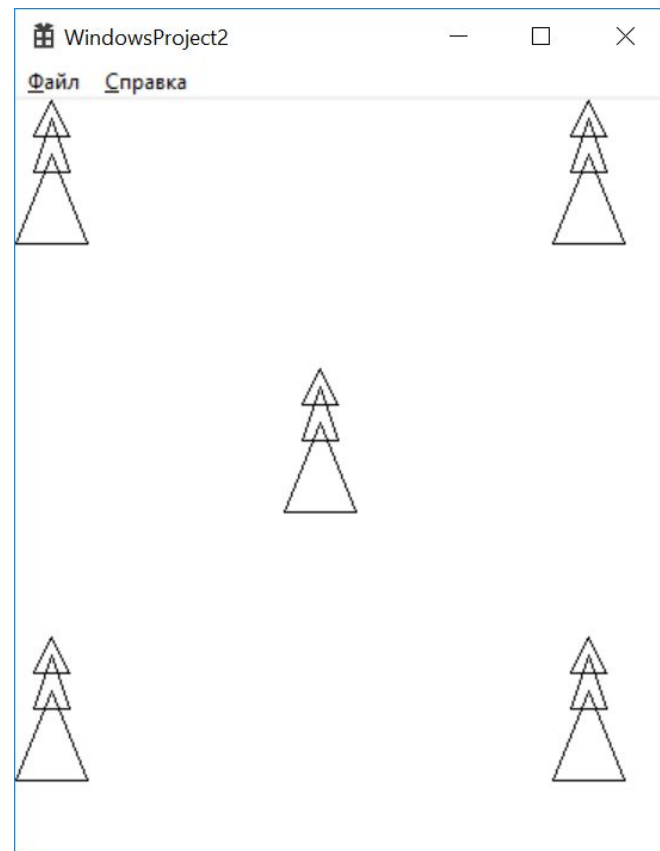
```
case WM_PAINT:  
{  
    PAINTSTRUCT ps;  
    HDC hdc = BeginPaint(hWnd, &ps);  
    // TODO: Добавьте сюда любой код прорисовки, использующий HDC...  
  
    StClausAuto(hdc, 60, 40);  
    StClausAuto(hdc, 0, 0);  
    StClausAuto(hdc, 200, 0);  
  
    EndPaint(hWnd, &ps);  
}  
break;
```

Задача 1.1: 5 логотипов по углам и в центре

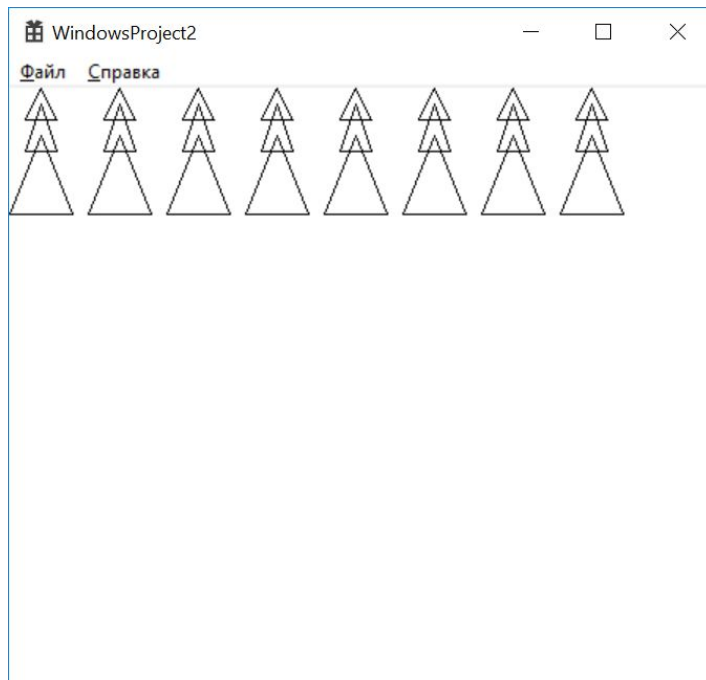


Задача 1.1: 5 логотипов по углам и в центре

```
247 case WM_PAINT:
248 {
249     PAINTSTRUCT ps;
250     HDC hdc = BeginPaint(hWnd, &ps);
251     // TODO: Добавьте сюда любой код прорисовки, использующий HDC...
252
253     StClausAuto(hdc, 0, 0);
254     StClausAuto(hdc, 200, 200);
255     StClausAuto(hdc, 100, 100);
256     StClausAuto(hdc, 0, 200);
257     StClausAuto(hdc, 200, 0);
258
259     EndPaint(hWnd, &ps);
260 }
261 break;
```



Задача 1.2: 8 логотипов в горизонтальную линию



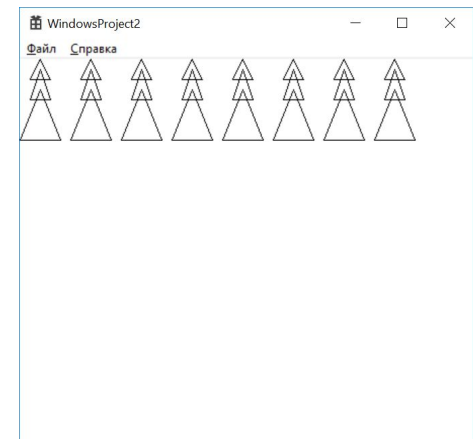
Задача 1.2: 8 логотипов в горизонтальную линию

Решение

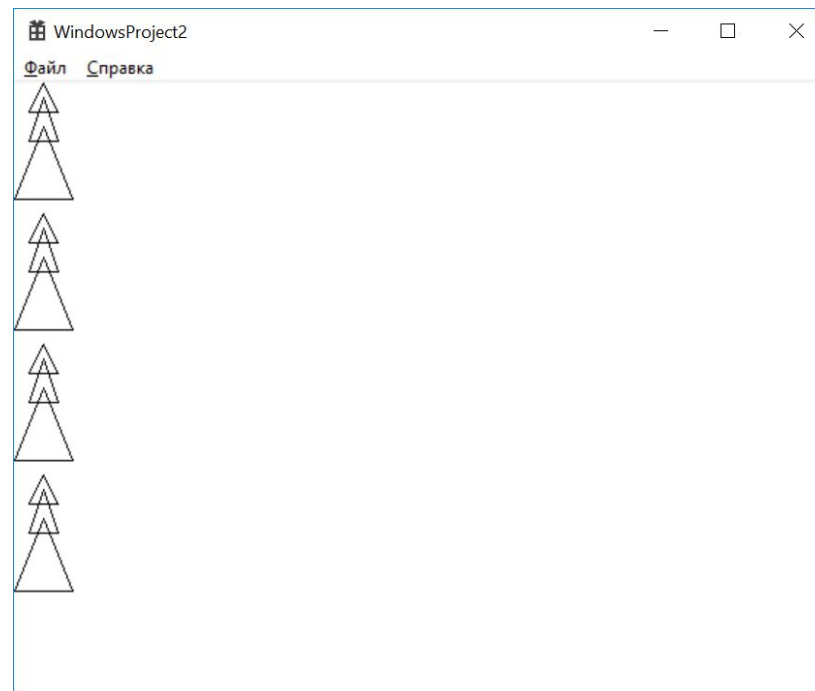
```
case WM_PAINT:
{
    PAINTSTRUCT ps;
    HDC hdc = BeginPaint(hWnd, &ps);
    // TODO: Добавьте сюда любой код прорисовки, использующий HDC...

    int x = 0;
    do {
        StClausAuto(hdc, x, 0);
        x += 50;
    } while (x < 400);

    EndPaint(hWnd, &ps);
}
break;
```



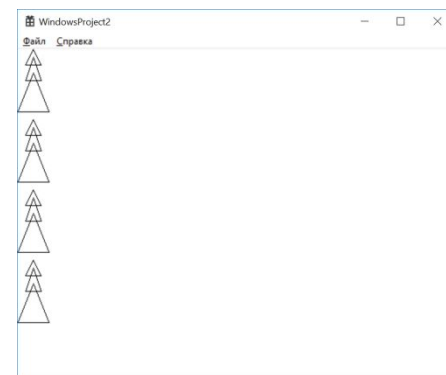
Задача 1.3: 4 логотипа в вертикальную линию



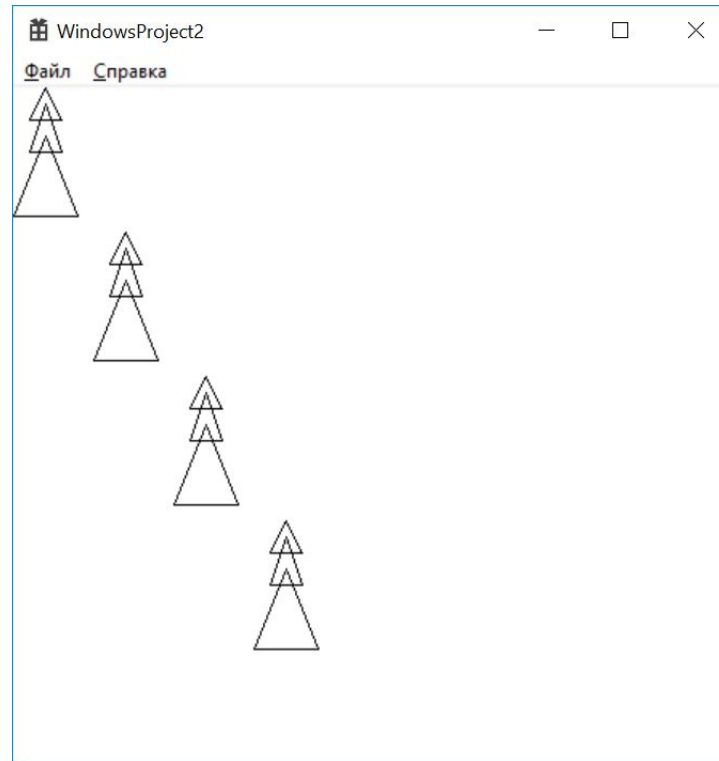
Задача 1.3 - РЕШЕНИЕ:

4 логотипа в вертикальную линию

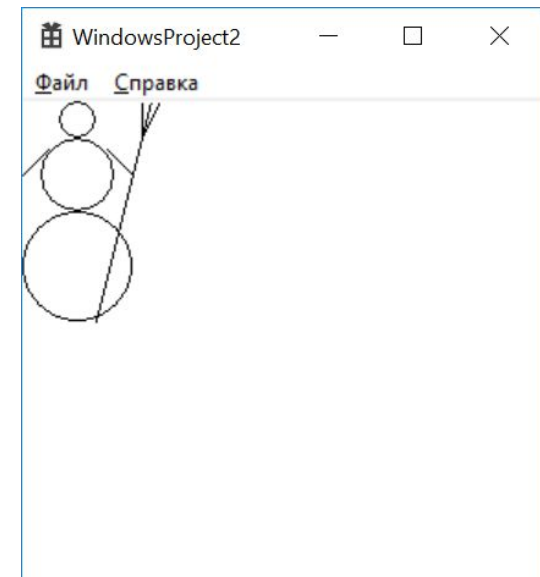
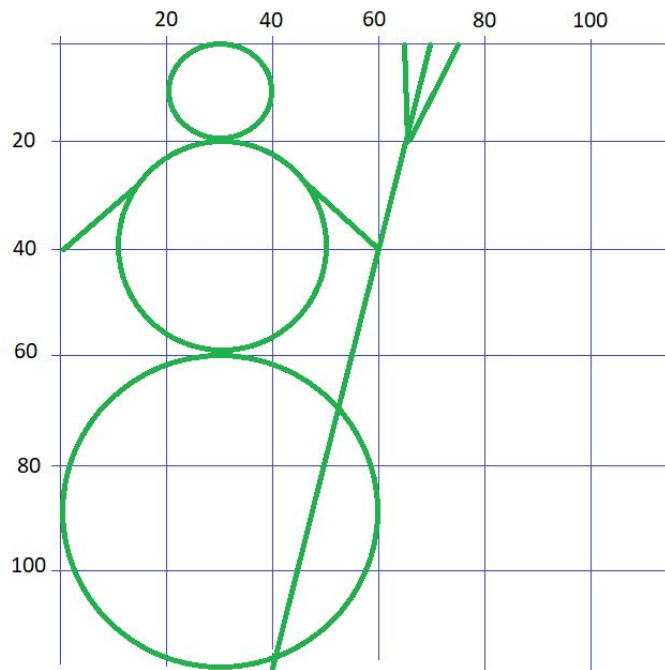
```
int y = 0;  
do {  
    StClausAuto(hdc, 0, y);  
    y += 90;  
} while (y < 300);
```



Задача 1.4: 4 логотипа в диагональную линию



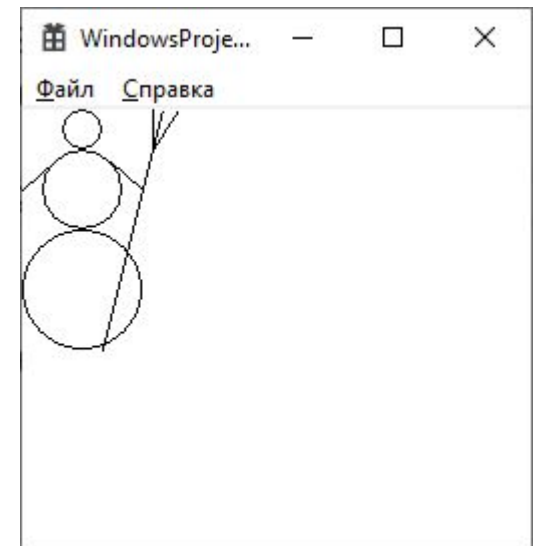
Задача 2.1 – Создать картинку по образцу



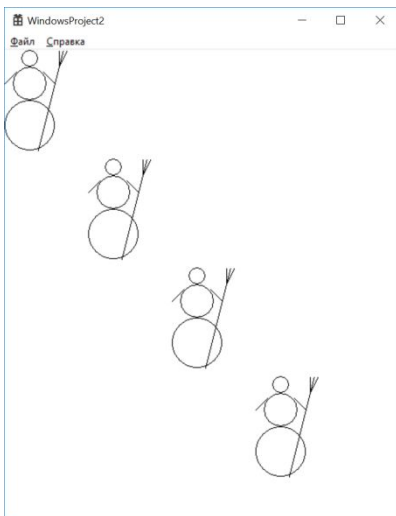
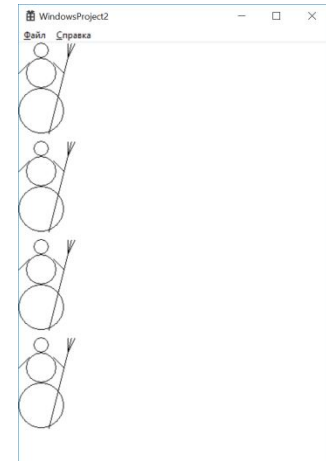
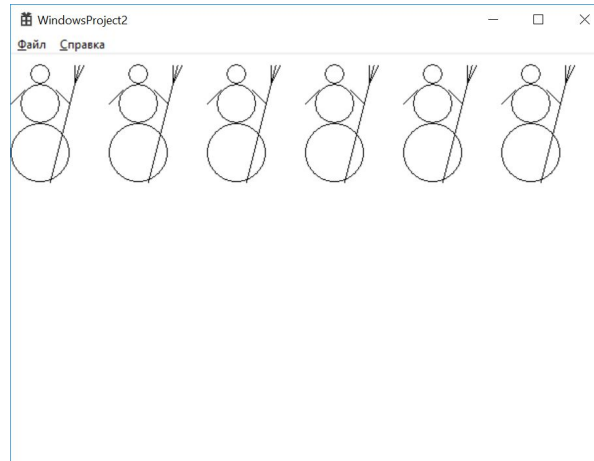
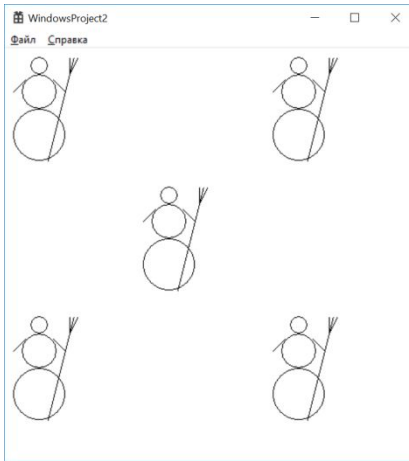
Задача 2.2 – Создать функцию SnowWoman

```
209 void SnowWoman(HDC hdc, int x, int y) {  
210     Ellipse(hdc, 20 + x, 0 + y, 40 + x, 20 + y);  
211     Ellipse(hdc, 10 + x, 20 + y, 50 + x, 60 + y);
```

```
261 case WM_PAINT:  
262 {  
263     PAINTSTRUCT ps;  
264     HDC hdc = BeginPaint(hWnd, &ps);  
265     // TODO: Добавьте сюда любой код прорисовки, использующий HDC...  
266  
267     SnowWoman(hdc, 0, 0);  
268  
269     EndPaint(hWnd, &ps);  
270 }  
271 break;
```

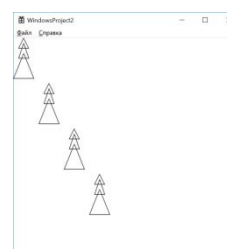
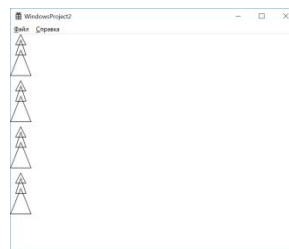
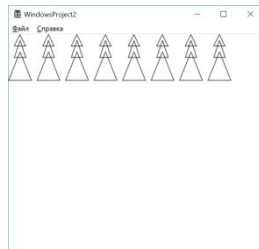
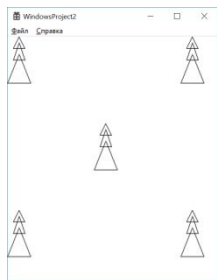


Задача 2.3 – 2.6 – Создать 4 рисунка из снежных баб



Домашнее задание

- 1) Доделать рисунки из Задач 1 и 2, которые не успели сделать на занятии в классе.
- 2) Нарисовать для всех циклических алгоритмов блоксхему.
- 3) Создать в виде отдельной функции логотип любого автомобиля. Созданная функция должна иметь вид `Logo(HDC hdc, int x, int y)`. При этом сделать логотип такого размера, чтобы он влезался по высоте окна не меньше 4 раз, по ширине не меньше 6 раз
- 4) Используя эту функцию создать узоры из логотипов по следующим 4 схемам:



- 5) Для последнего узора нарисовать блоксхему алгоритма.

Домашнее задание – оформление

Для сдачи домашней работы нужно иметь:

- 1) код программы
- 2) Расчет картинки – на бумаге (на отдельном листе или в тетради) или в файле с видимыми признаками расчетов
- 3) Скрин шот выполнения программы
- 4) Блоксхемы для циклических алгоритмов

Срок выполнения – до следующей встречи на лабораторной работе

Если болел/не мог – это становится «долгом»