

Основы программирования

ФИСТ 1 курс

Власенко

Олег

Федосович

Лекция 3

Создание собственных функций для рисования по заданным
координатам.

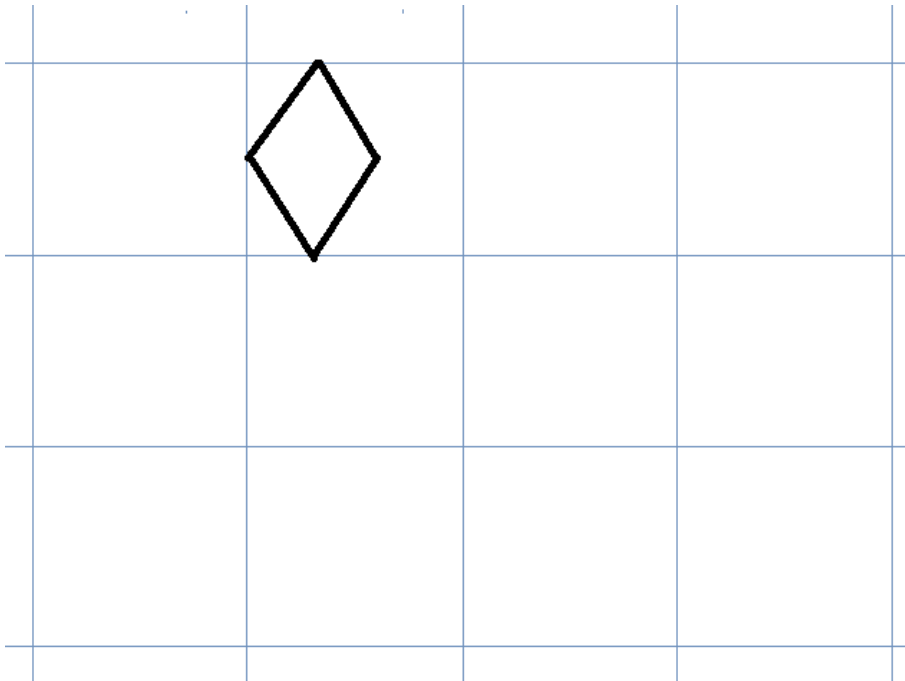
Тригонометрия на службе графики.

Трассировка циклической программы

Домашняя работа

1. Прочитать про рисование в WinAPI
<http://radiofront.narod.ru/htm/prog/htm/winda/api/paint.html>
2. Установить Visual Studio (если ранее не установили).
3. Создать простейшее Win32 приложение
4. Нарисовать домик и танк в этом приложении
(вместо танка можно нарисовать что-то более жизнеутверждающее - с эллипсами или дугами)

Ромб



// Рисуем ромб

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

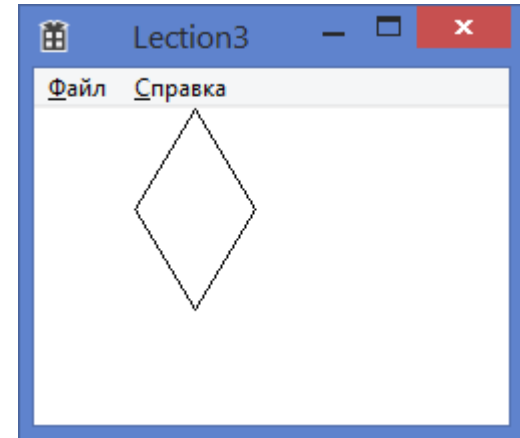
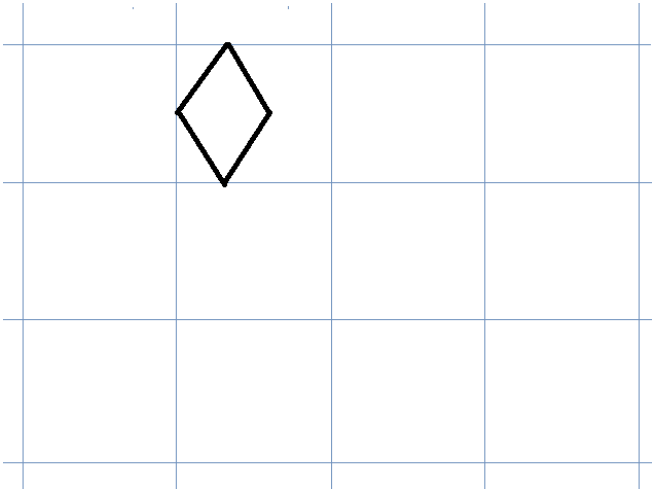
```
LineTo(hdc, 50, 50);
```

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

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

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

Ромб



// Рисуем ромб

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

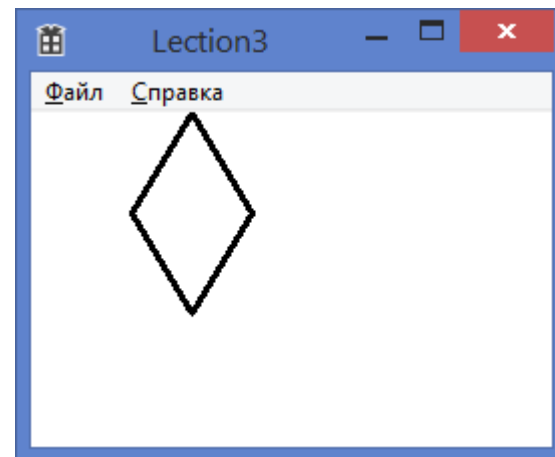
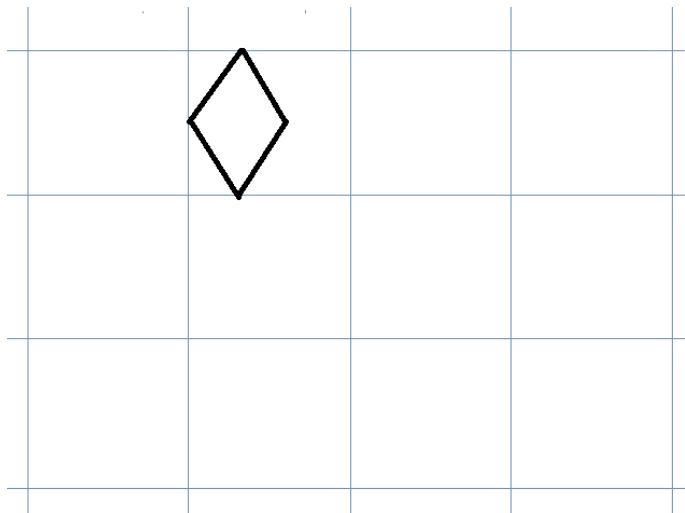
```
LineTo(hdc, 50, 50);
```

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

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

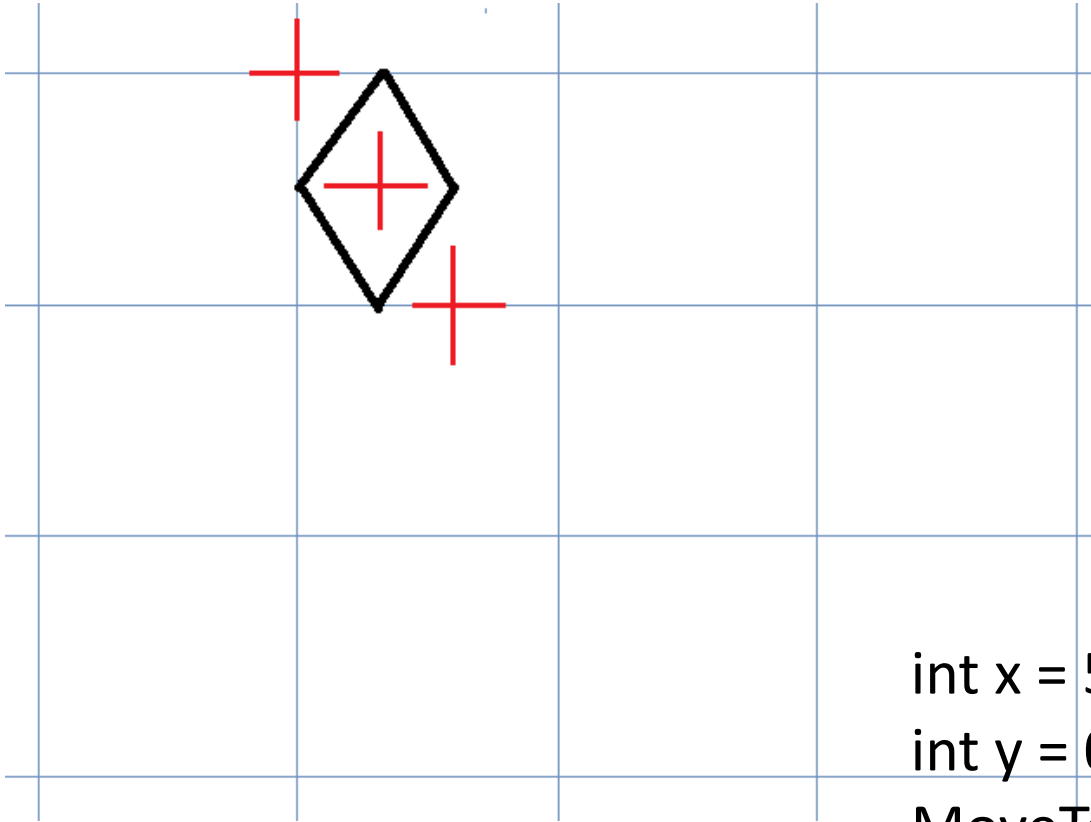
```
LineTo(hdc, 80, 0);
```

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);
```

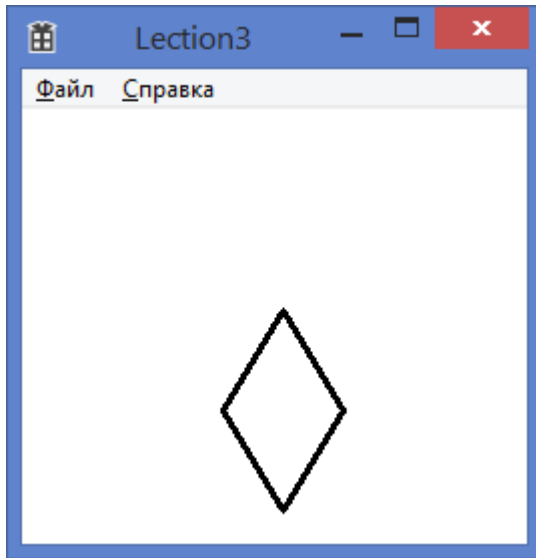
```
LineTo(hdc, x, y + 50);
```

```
LineTo(hdc, x + 30, y + 100);
```

```
LineTo(hdc, x + 60, y + 50);
```

```
LineTo(hdc, x + 30, y);
```

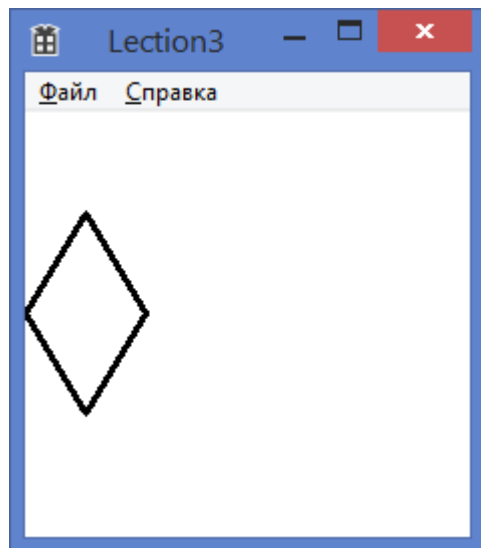
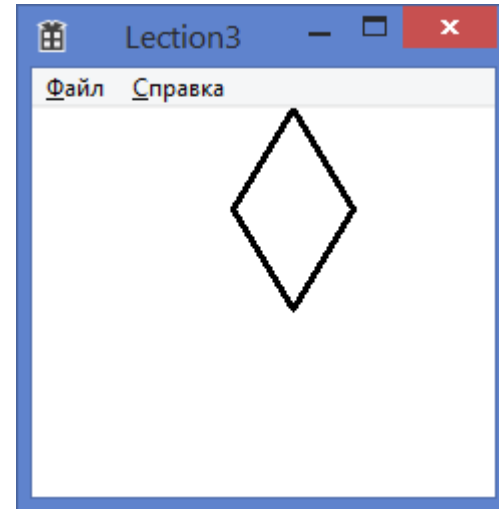
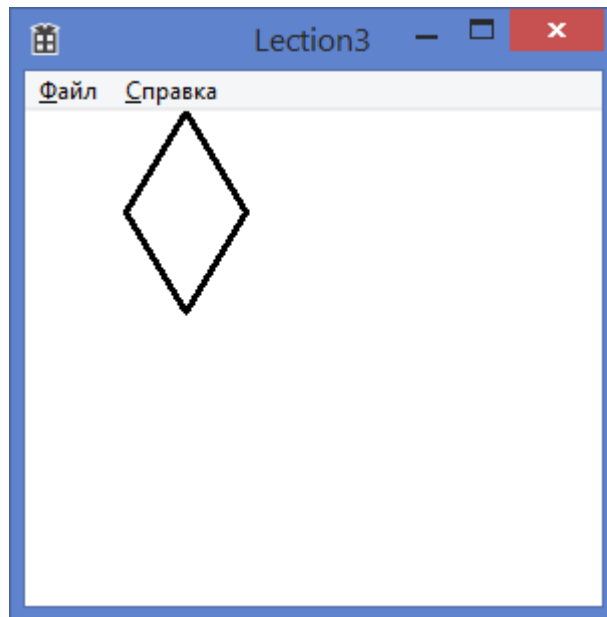
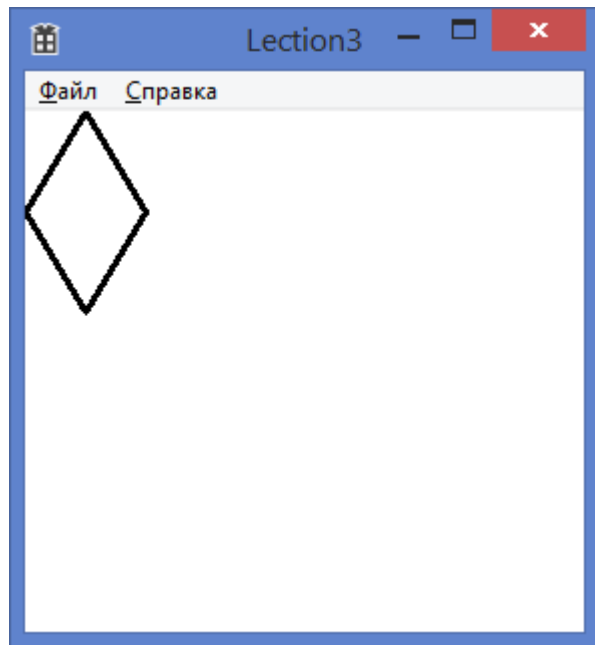
Относительные координаты



```
int x = 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);
```

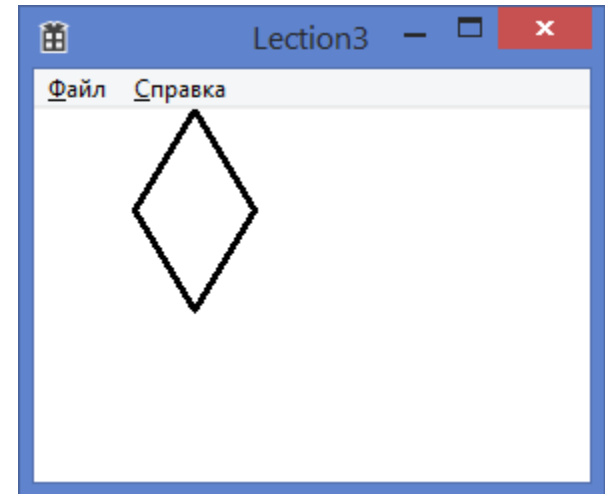
```
int x = 150;  
int y = 20;  
MoveToEx(hdc, x + 30, y, NULL);  
LineTo(hdc, x, y + 50);  
LineTo(hdc, x + 30, y + 100);  
LineTo(hdc, x + 60, y + 50);  
LineTo(hdc, x + 30, y);
```

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);
```

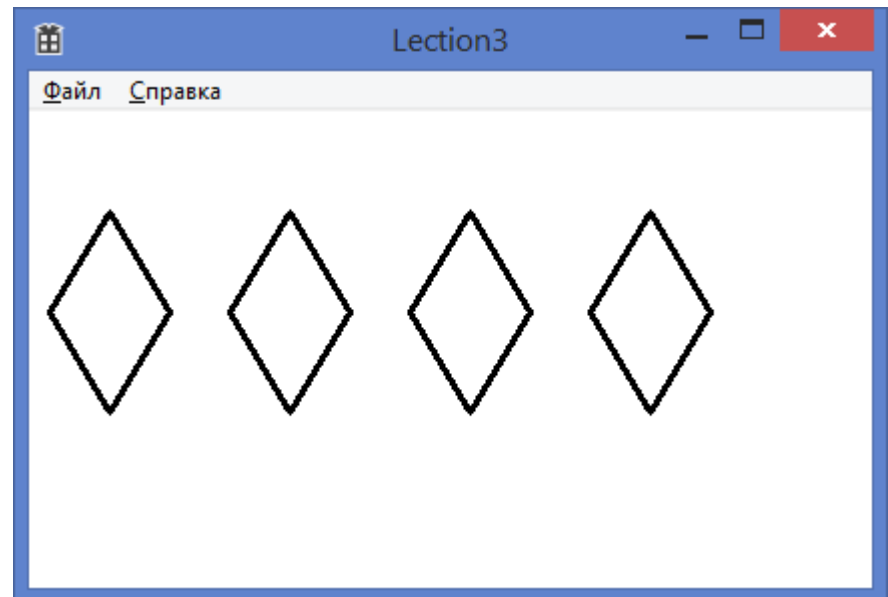
...

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 – при помощи цикла

...

```
HPEN hPen;
```

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

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

```
int x = 10;
```

```
int y = 50;
```

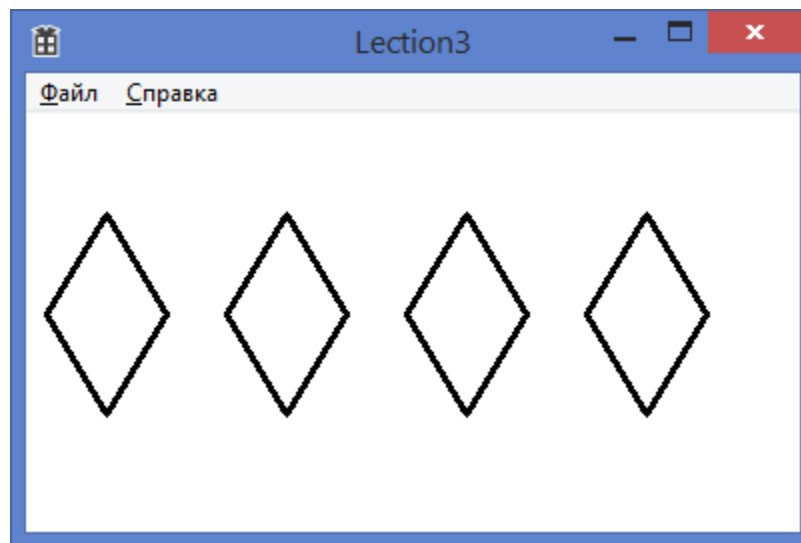
```
do {
```

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

```
    x += 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;
```

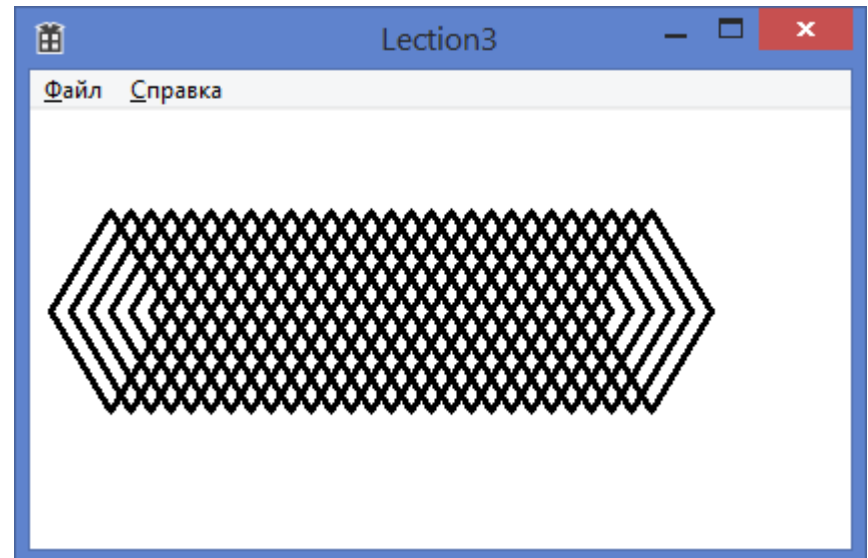
```
do {
```

```
    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;
```

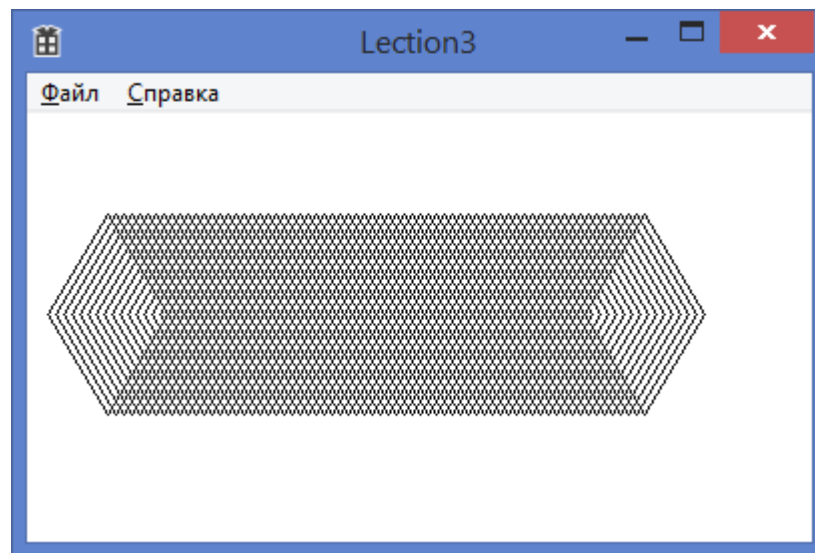
```
do {
```

```
    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;
```

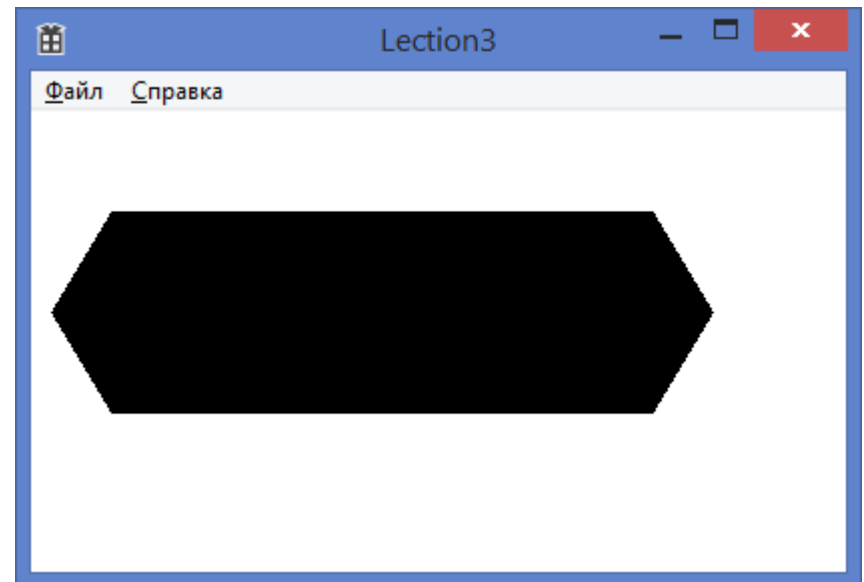
```
int y = 50;
```

```
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;
```

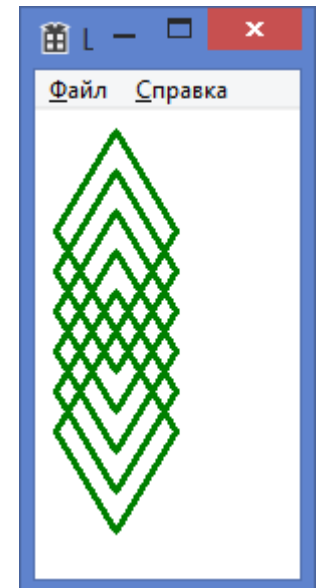
```
int y = 10;
```

```
do {
```

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

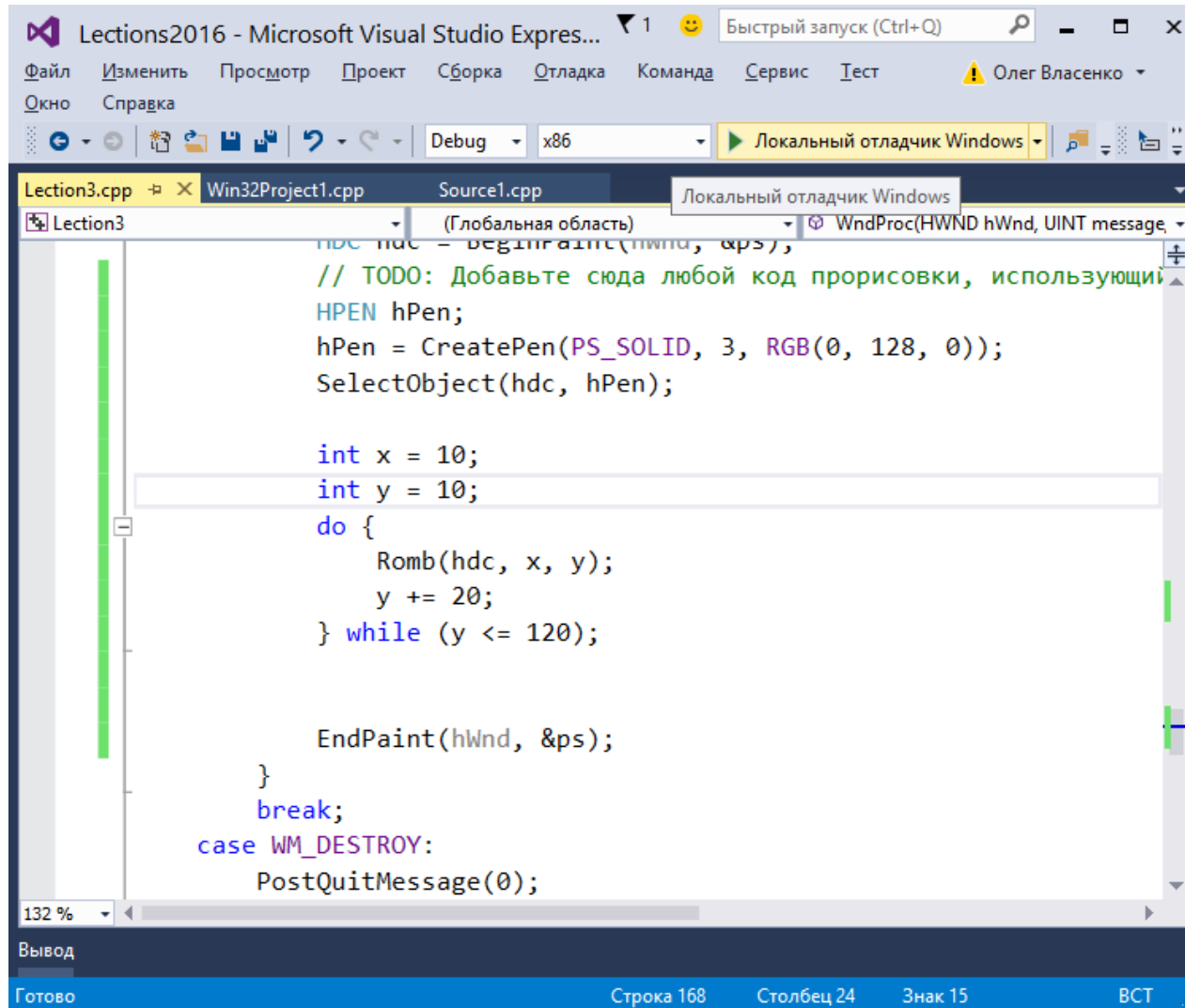
```
    y += 20;
```

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

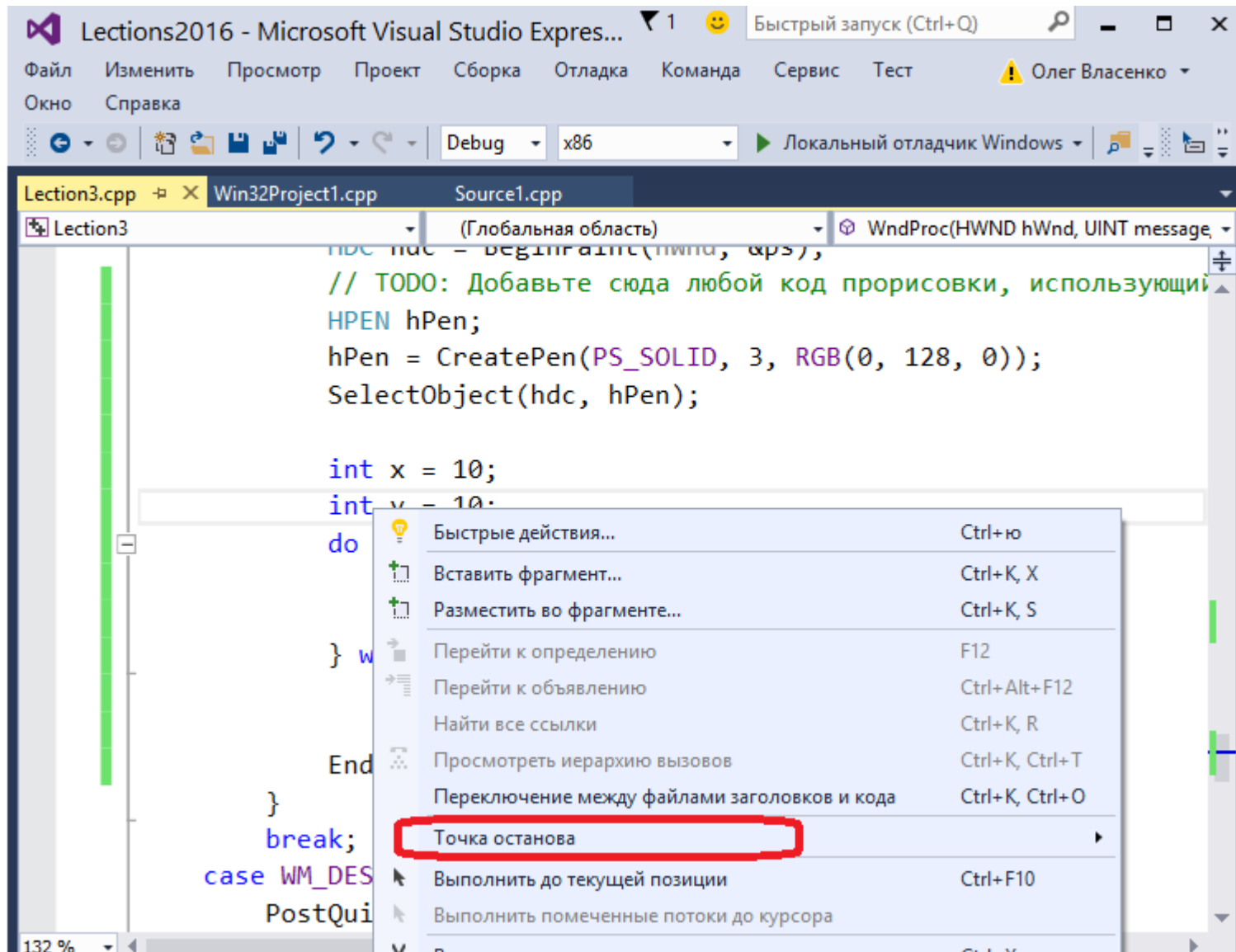


Трассировка циклического алгоритма (2)

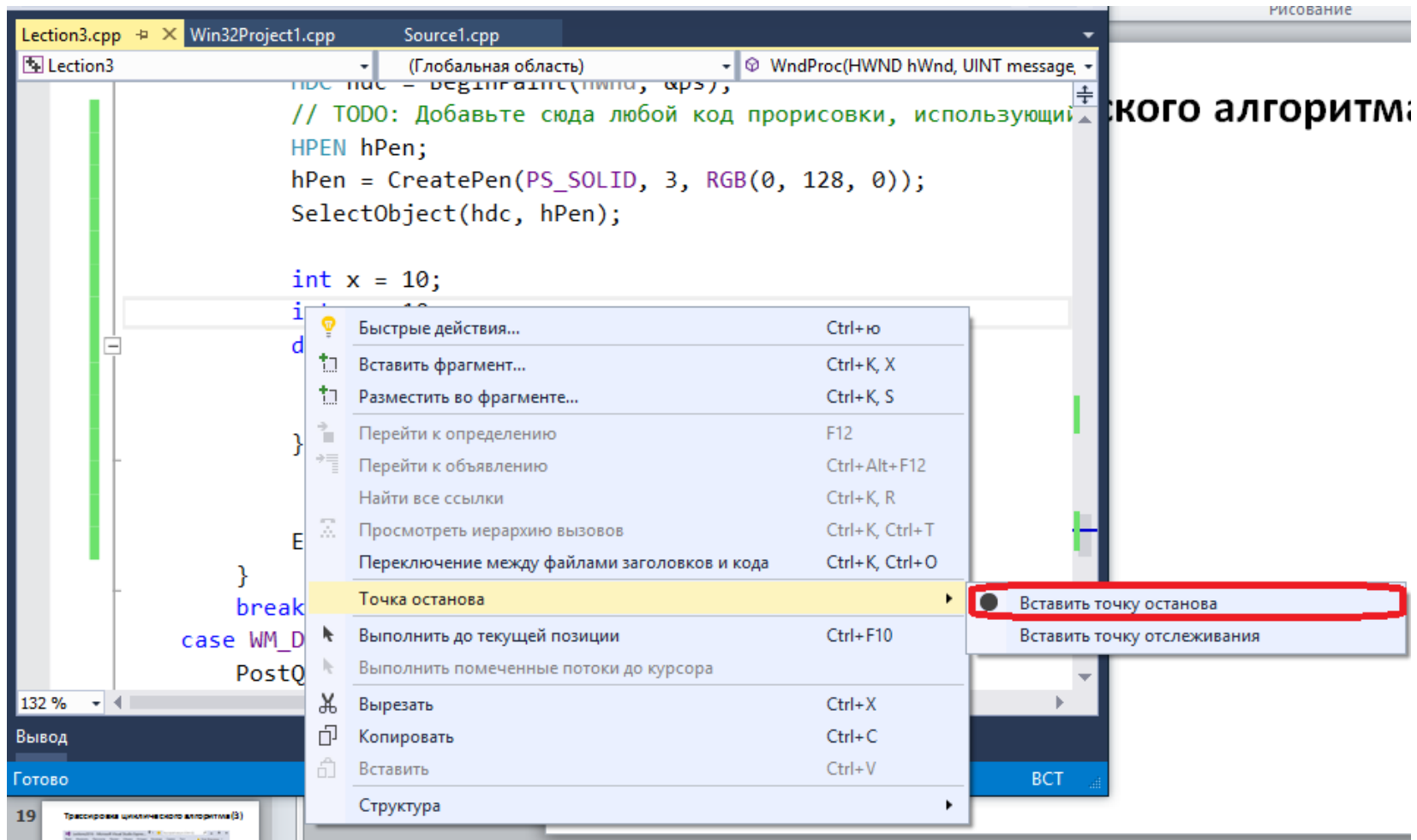
Установка точки останова



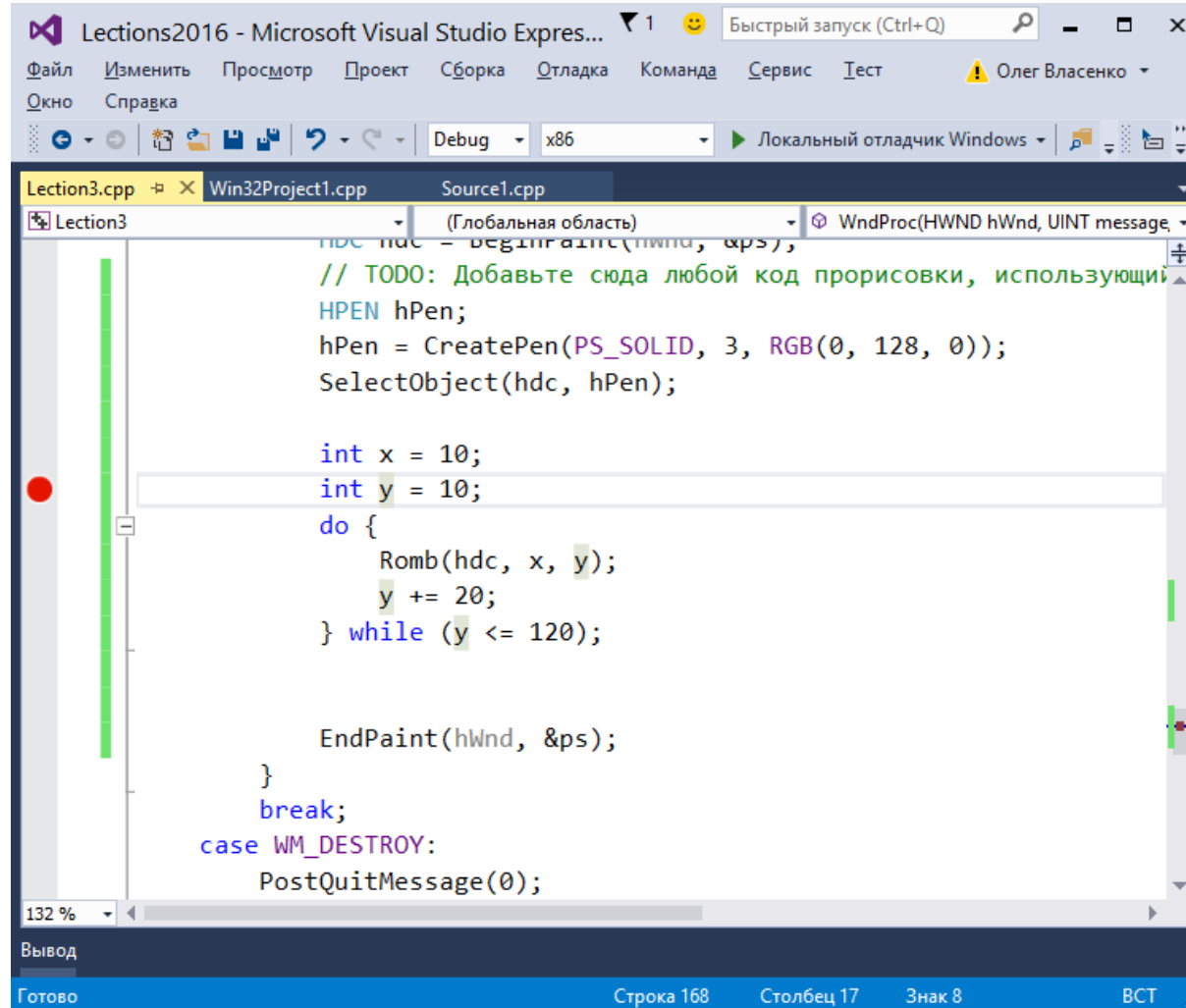
Трассировка циклического алгоритма (3)



Трассировка циклического алгоритма (4)



Трассировка циклического алгоритма (5)



```
Microsoft Visual Studio Express... 1 Быстрый запуск (Ctrl+Q)
Файл Изменить Просмотр Проект Сборка Отладка Команда Сервис Тест Олер Власенко
Окно Справка
Debug x86 Локальный отладчик Windows
Lecture3.cpp Win32Project1.cpp Source1.cpp
Lecture3 (Глобальная область) WndProc(HWND hWnd, UINT message,
hdc 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);
    y += 20;
} while (y <= 120);

EndPoint(hWnd, &ps);
break;
case WM_DESTROY:
    PostQuitMessage(0);
```

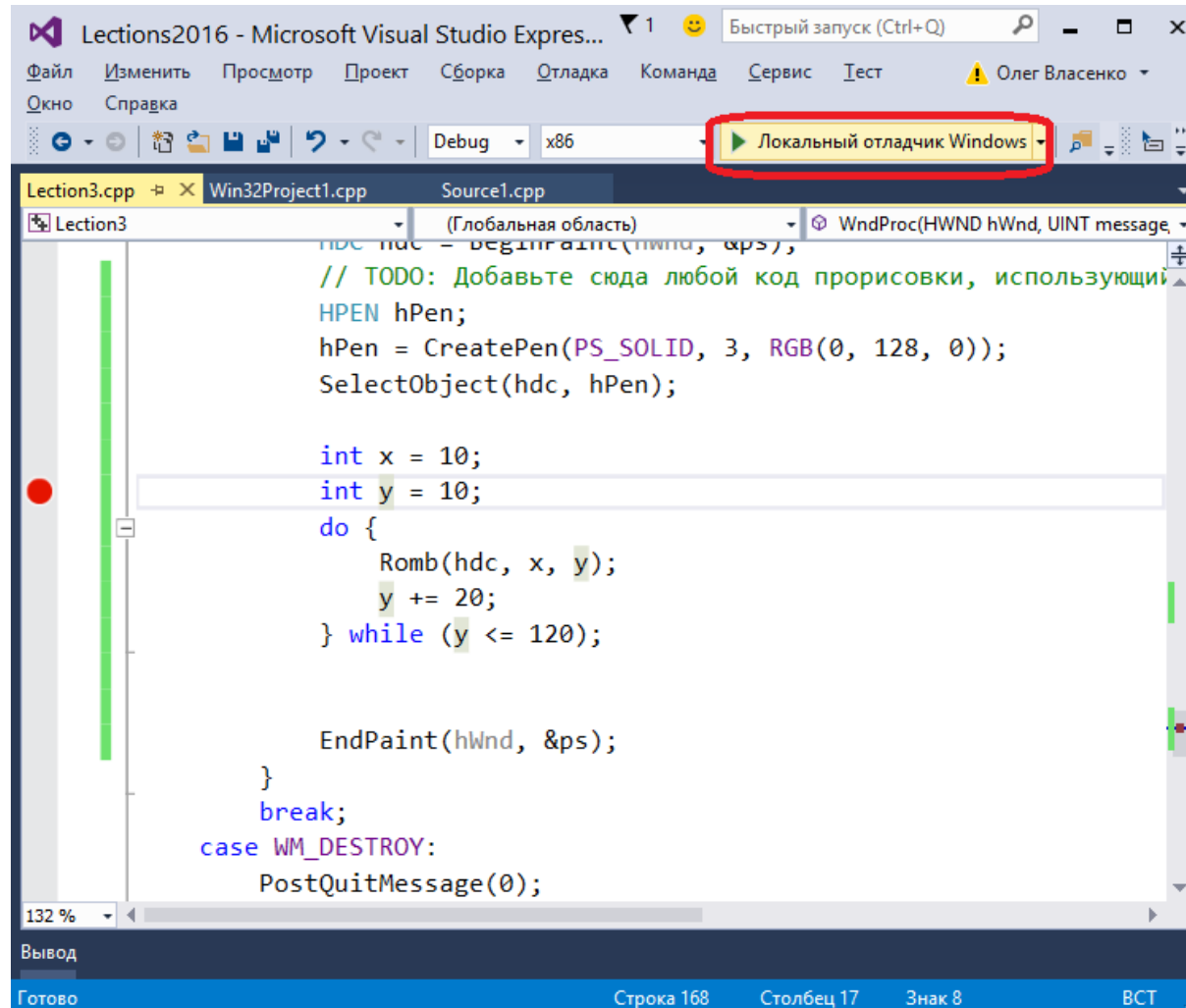
132 %

Вывод

Готово Строка 168 Столбец 17 Знак 8 ВСТ

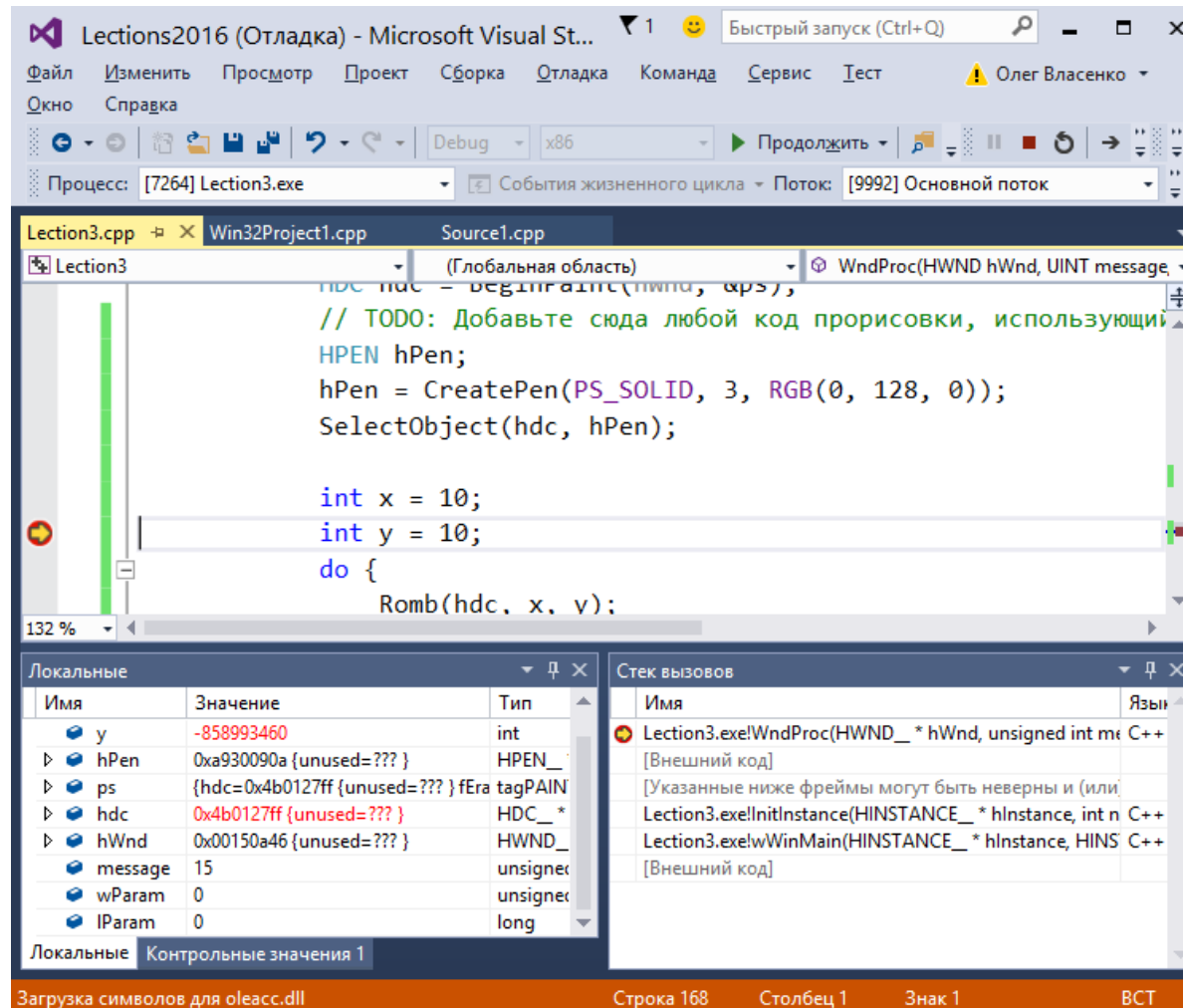
Трассировка циклического алгоритма (6)

Запуск трассировки



Трассировка циклического алгоритма (7)

(при)остановка в точке останова



Visual Studio interface showing the debugger paused at a breakpoint in Lesson3.cpp. The code is in the Global scope, and the function is WndProc(HWND hWnd, UINT message). The breakpoint is set at the start of the do-while loop.

```
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);  
}
```

Locals window (Локальные):

Имя	Значение	Тип
y	-858993460	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
lParam	0	long

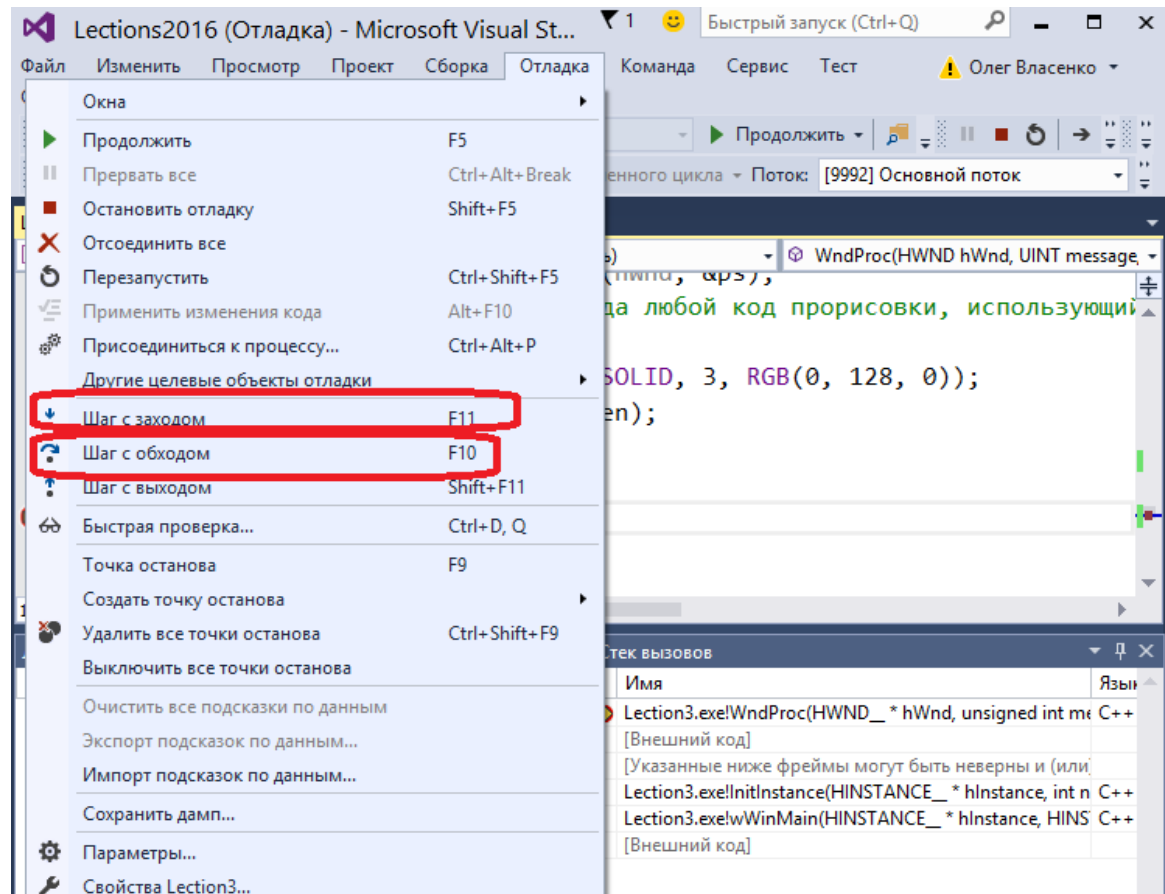
Call Stack window (Стек вызовов):

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 lpCmdLine, int nCmdShow)	C++
[Внешний код]	

Bottom status bar: 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)

Visual Studio interface showing the execution of a program. The main window displays the source code of `Lesson3.cpp` with the following code:

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

EndPaint(hWnd, &ps);
```

The debugger is currently at the `do` statement of the `while` loop. The `Locals` window shows the current state of variables:

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

The `Call Stack` window shows the sequence of function calls:

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]	

At the bottom, the status bar shows: Загрузка символов для oleacc.dll, Строка 170, Столбец 1, Знак 1, ВСТ.

Трассировка циклического алгоритма (10)

Еще шаг сделан (F10)

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; ≤ 1 мс прошло
} while (y <= 120);
```

132 %

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 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]	

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

Процесс: [7264] Lection3.exe События жизненного цикла Поток: [9992] Основной поток

Lection3.cpp Win32Project1.cpp Source1.cpp

Lection3 (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
Lection3.exe!WndProc(HWND_ * hWnd, unsigned int m...	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или]	
Lection3.exe!InitInstance(HINSTANCE_ * hInstance, int n...	C++
Lection3.exe!wWinMain(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

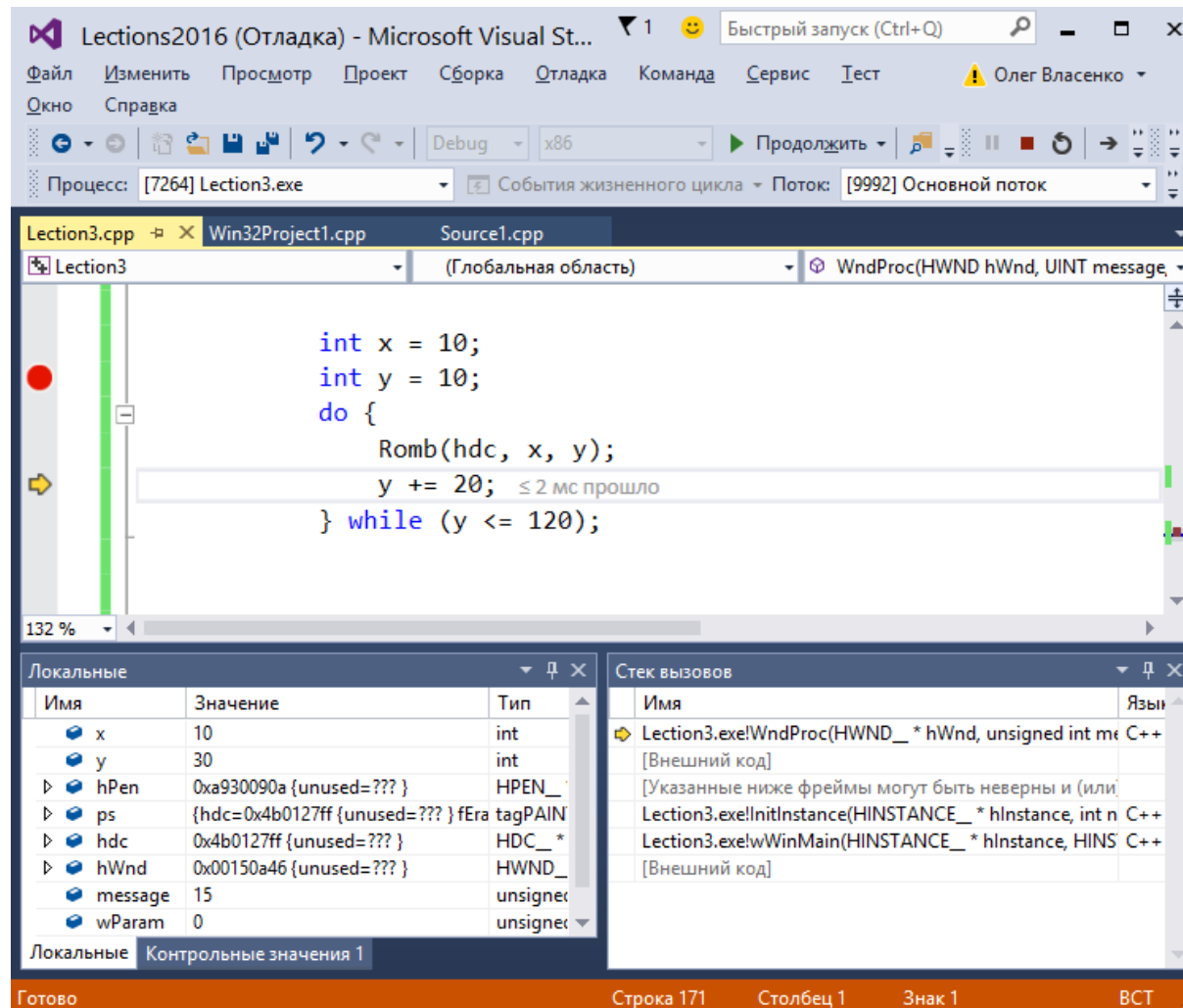
Локальные Контрольные значения 1

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



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 %

Имя	Значение	Тип
x	10	int
y	30	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

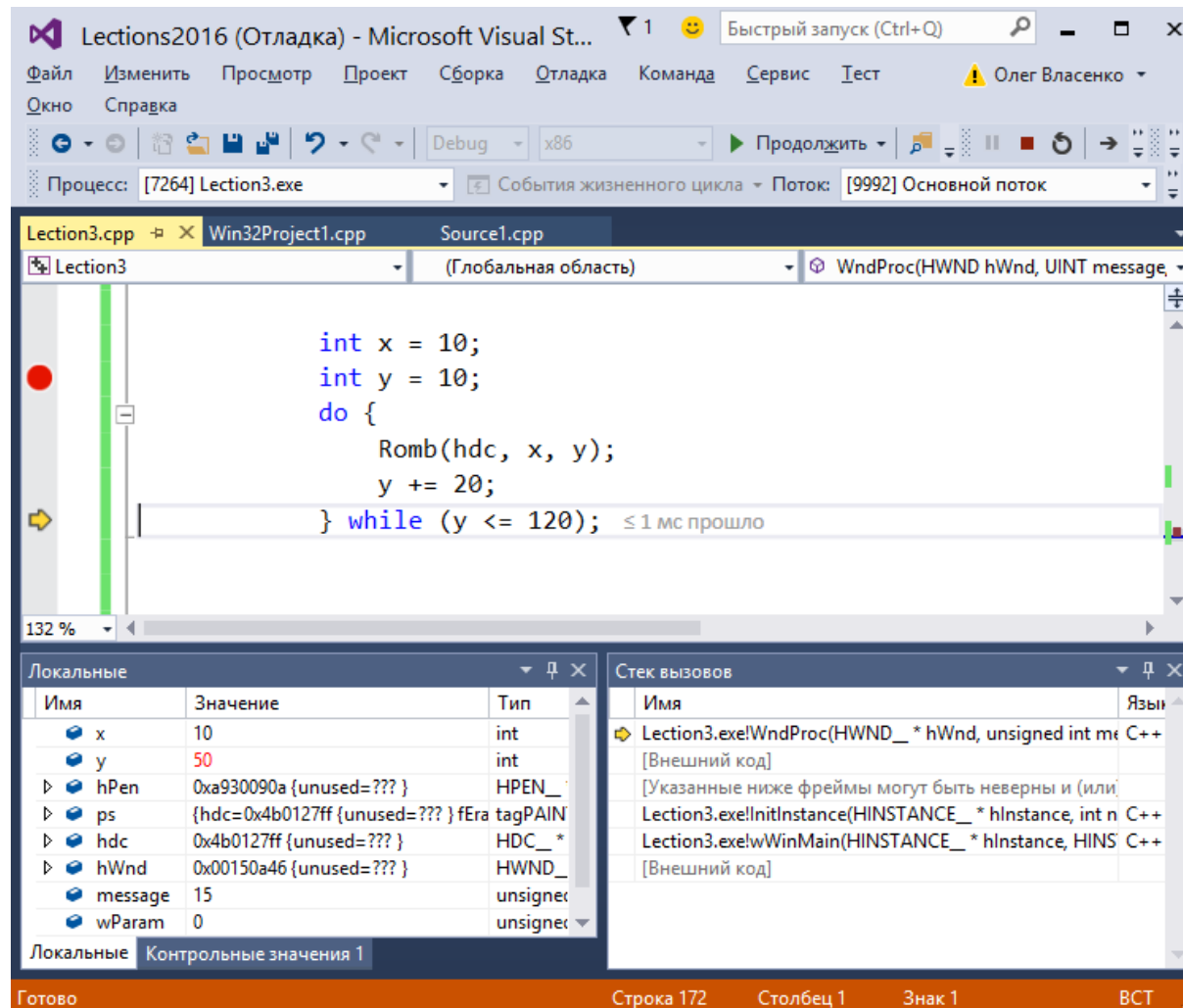
Локальные Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND __hWnd, unsigned int message)	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или) отсутствовать]	
Lesson3.exe!InitInstance(HINSTANCE __hInstance, int nCmdShow)	C++
Lesson3.exe!wWinMain(HINSTANCE __hInstance, HINSTANCE __hPrevInstance, LPWSTR __lpCmdLine, int __iCmdShow)	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 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!WinMain(HINSTANCE __hInstance, HINSTANCE __hPrevInstance, LPSTR __lpCmdLine, int __iCmdShow)	C++
[Внешний код]	

Готово Строка 172 Столбец 1 Знак 1 ВСТ

Трассировка циклического алгоритма (15)

Еще шаг сделан

Lesson3.cpp Win32Project1.cpp Source1.cpp

Lesson3 (Глобальная область) WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)

```
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 tagPAINTSTRUCT}	PAINTSTRUCT
hdc	0x4b0127ff {unused=???}	HDC_*
hWnd	0x00150a46 {unused=???}	HWND_*
message	15	unsigned int
wParam	0	unsigned int

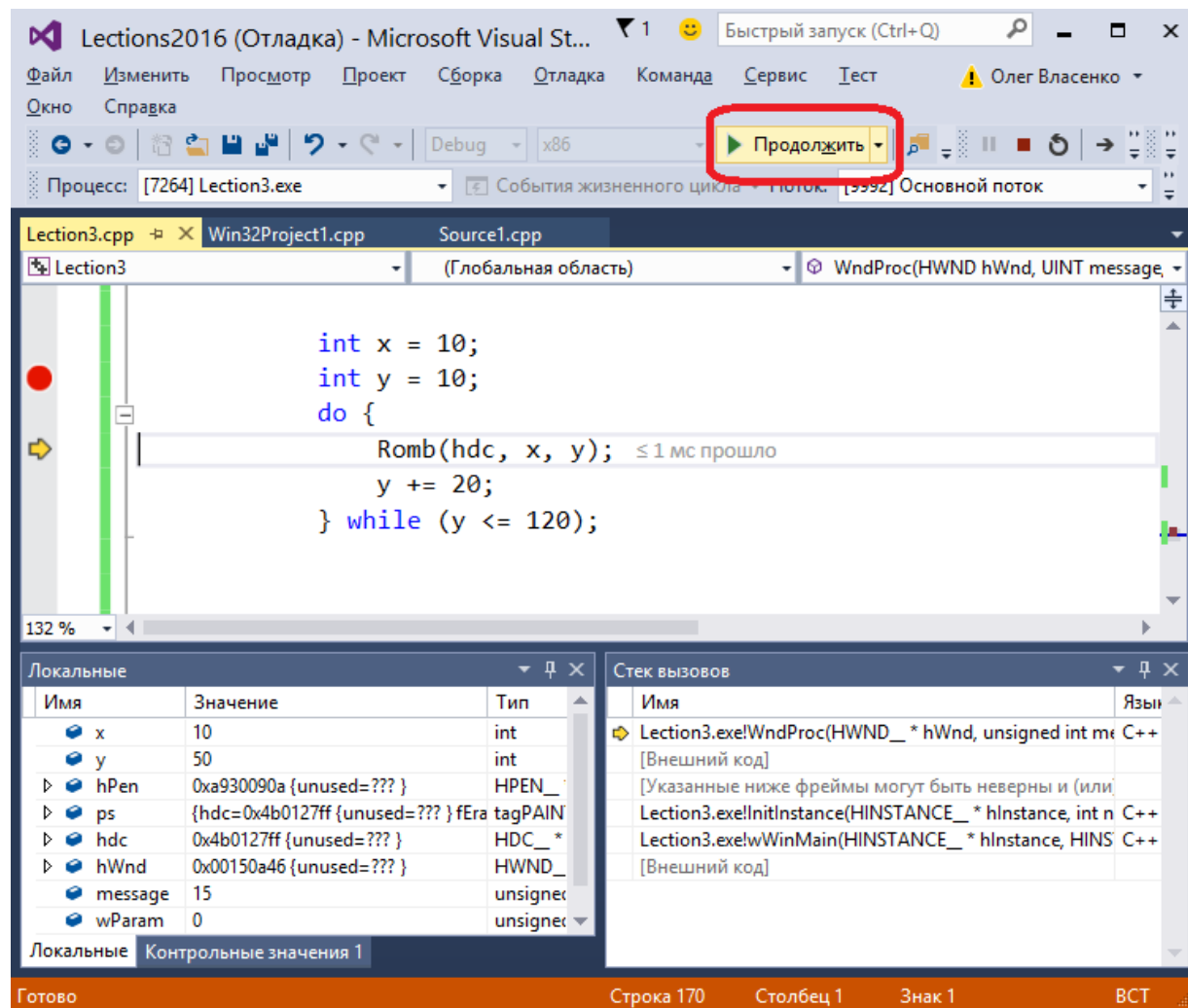
Локальные Контрольные значения 1

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int message, WPARAM wParam, LPARAM lParam)	C++
[Внешний код]	
[Указанные ниже фреймы могут быть неверны и (или) могут отсутствовать]	
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 (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 "События жизненного цикла:" dropdown shows "[9992] Основной поток". The code editor displays the following C++ code:

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

The "Локальные" (Locals) window at the bottom left shows the following variables and their values:

Имя	Значение	Тип
x	10	int
y	50	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

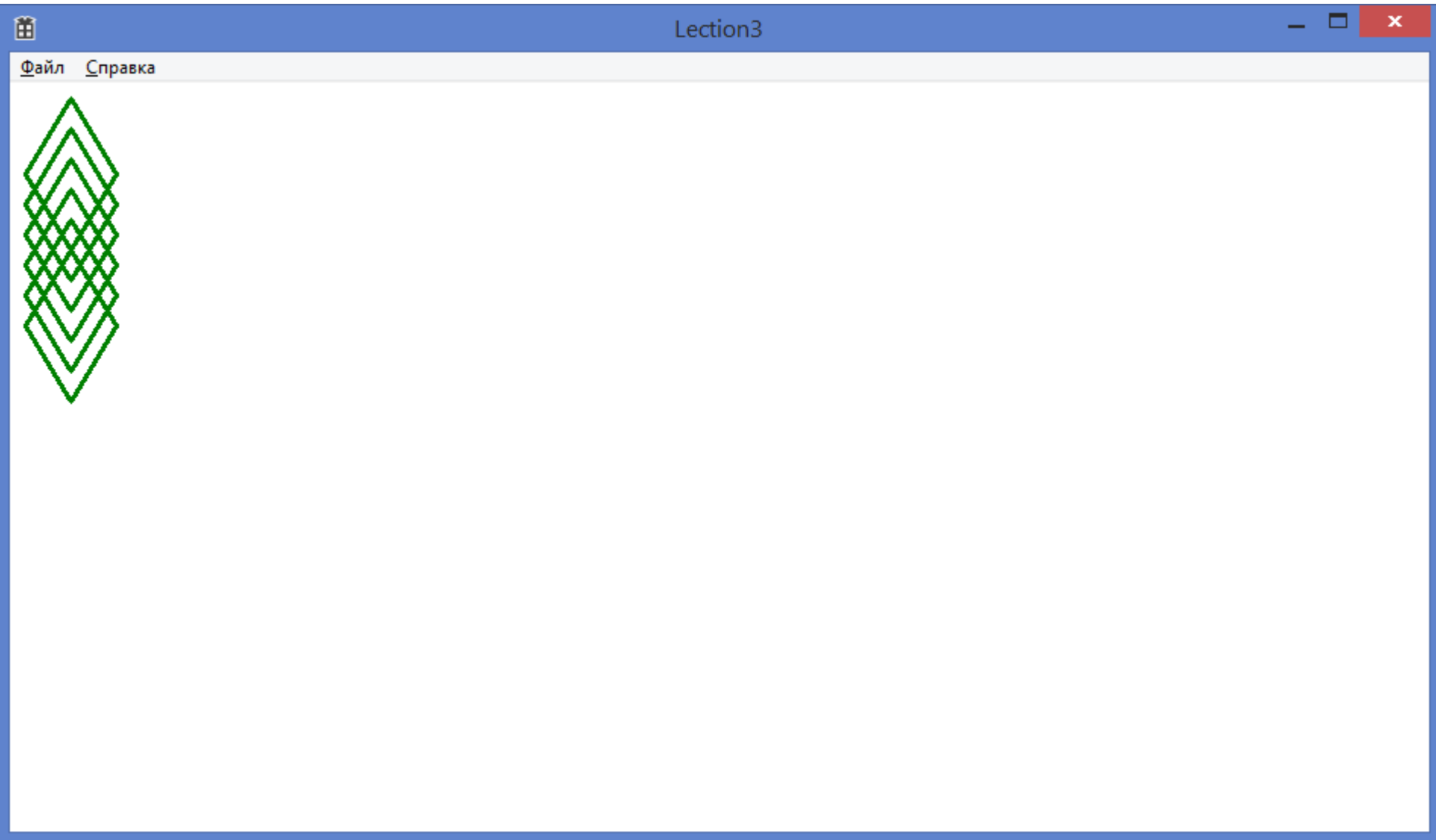
The "Стек вызовов" (Call Stack) window at the bottom right shows the following call stack:

Имя	Язык
Lesson3.exe!WndProc(HWND_ * hWnd, unsigned int message, WPARAM wParam)	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]	

The status bar at the bottom shows "Готово", "Строка 170", "Столбец 1", "Знак 1", and "ВСТ".

Трассировка циклического алгоритма (17)

Приложение работает!



Изменяем сразу и X и Y

...

```
int x = 10;
```

```
int y = 50;
```

```
do {
```

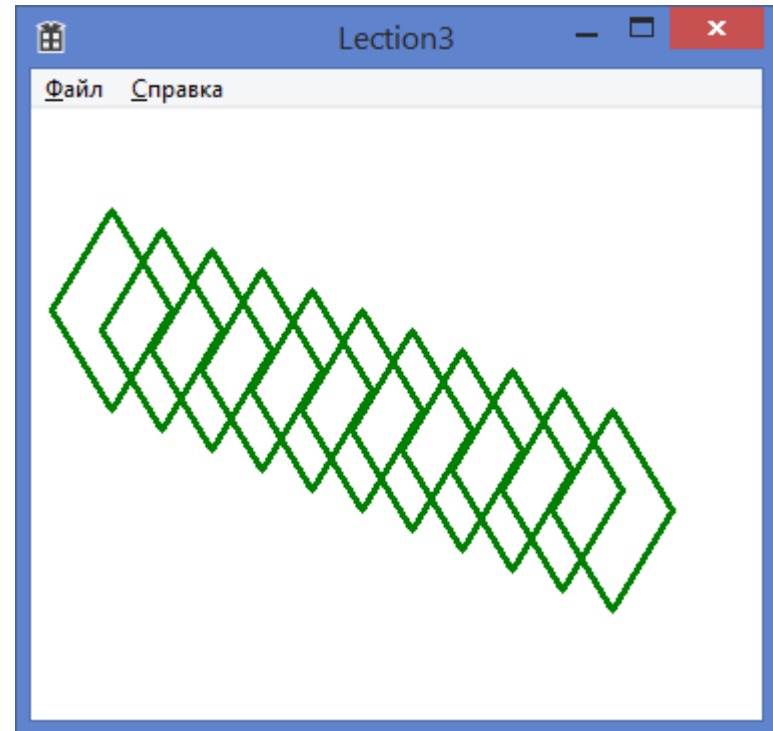
```
    Romb(hdc, x, y);
```

```
    x += 25;
```

```
    y += 10;
```

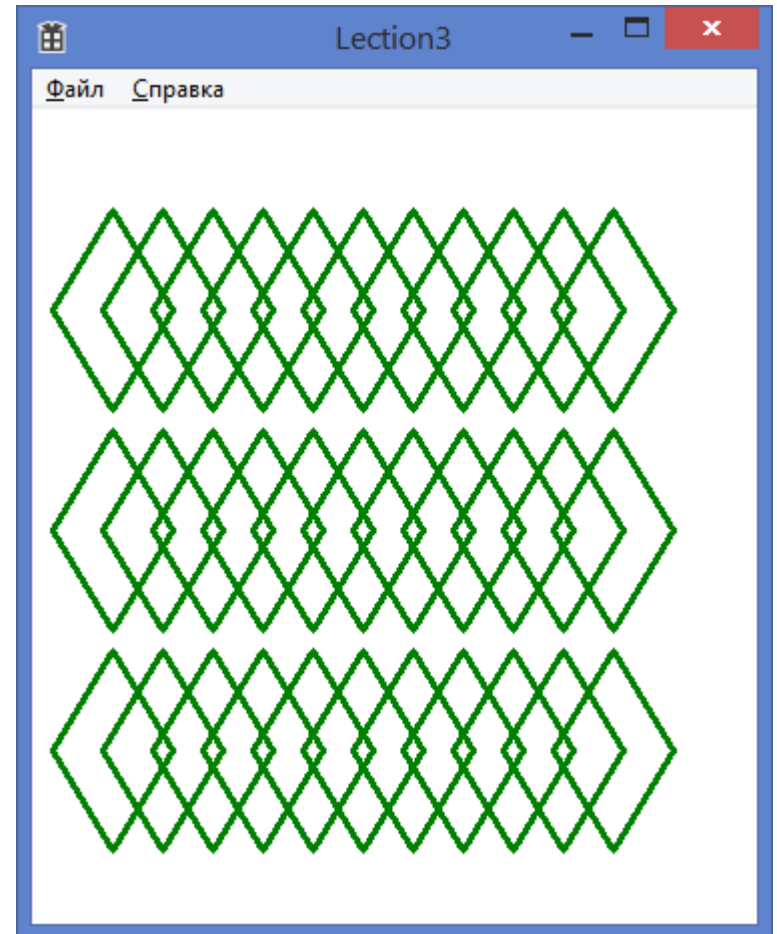
```
} while (x <= 280);
```

...



ВЛОЖЕННЫЕ ЦИКЛЫ

```
...  
int y = 50;  
do {  
    int x = 10;  
    do {  
        Romb(hdc, x, y);  
        x += 25;  
    } while (x <= 280);  
    y += 110;  
} while (y <= 280);  
...
```



ВЛОЖЕННЫЕ ЦИКЛЫ

...

```
int y = 50;
```

```
do {
```

```
    int x = 10;
```

```
    do {
```

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

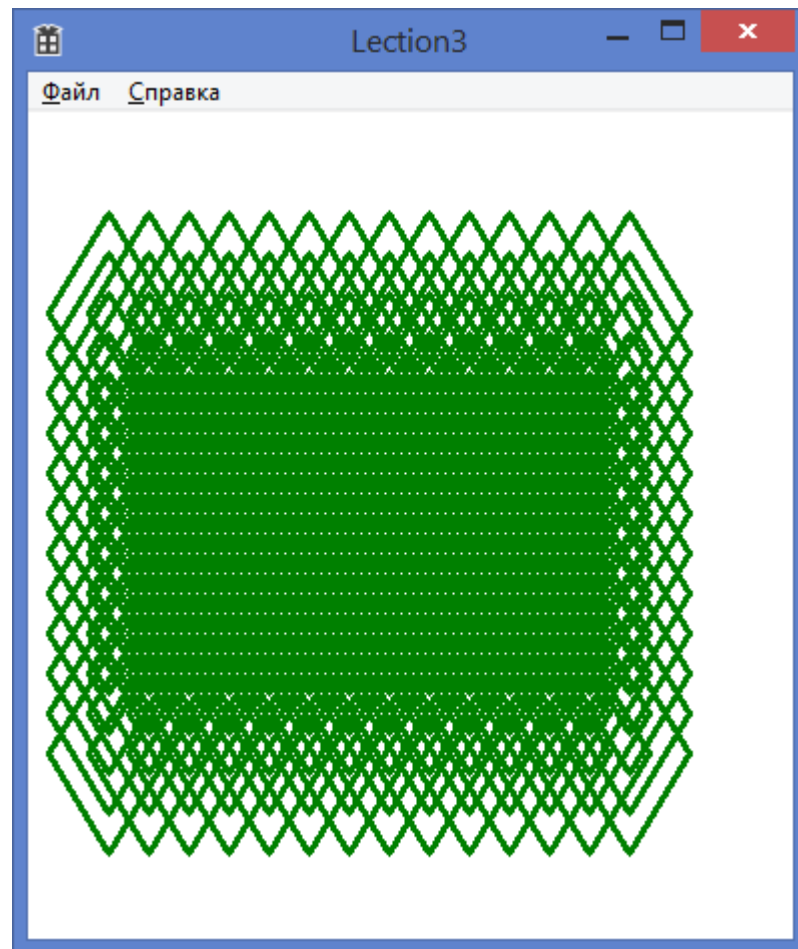
```
        x += 20;
```

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

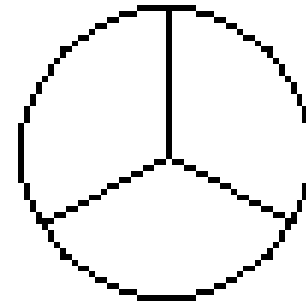
```
    y += 20;
```

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

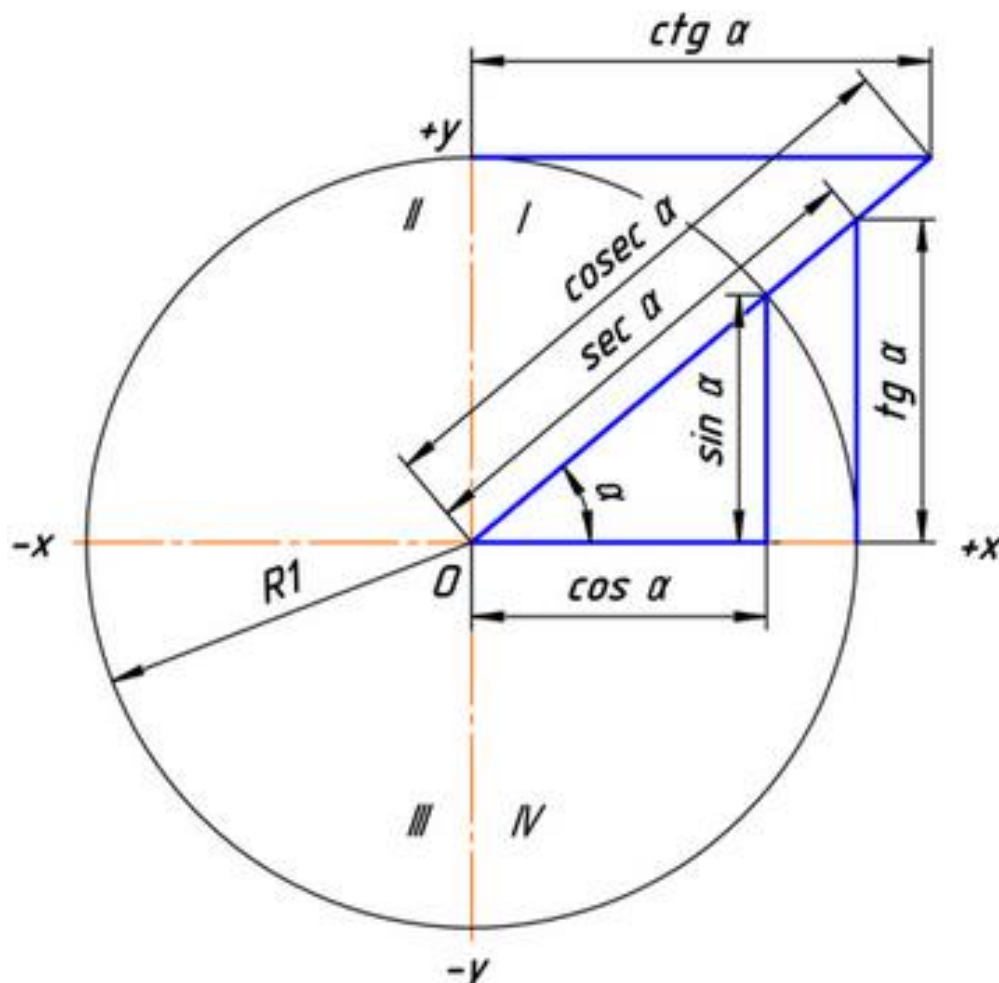
...



Логотип Mercedes-Benz



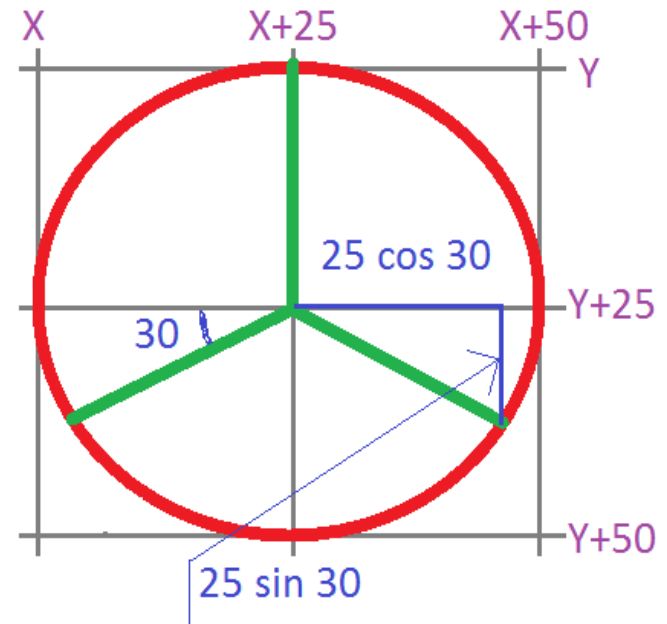
Вспоминаем тригонометрию



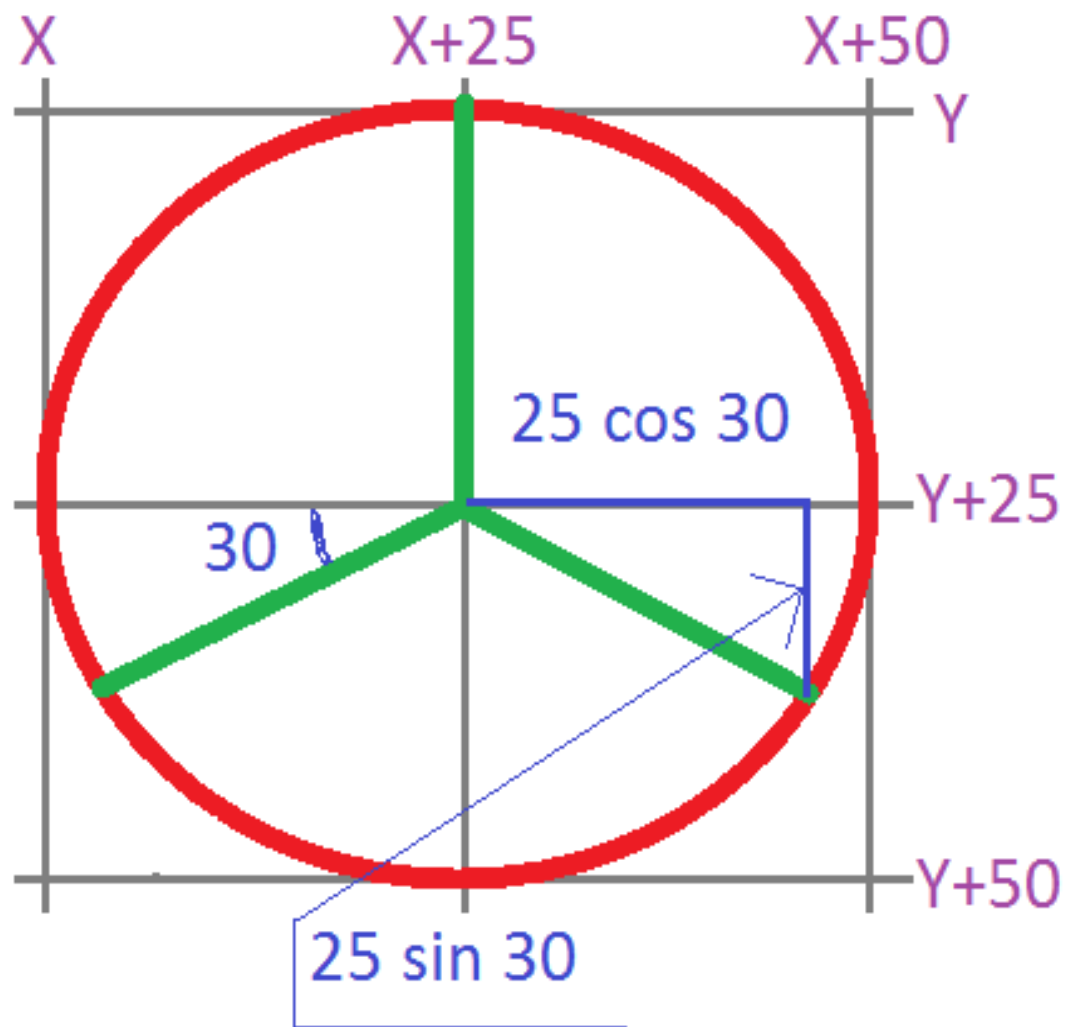
Численные значения тригонометрических функций угла в тригонометрической окружности с радиусом, равным единице

Функция отрисовки упрощенного логотипа МерседесБенц

```
void MercedesBenz(HDC hdc, int x, int y) {  
    Ellipse(hdc, x, y, x + 50, y + 50);  
    MoveToEx(hdc, x + 25, y + 25, NULL);  
    LineTo(hdc, x + 25, y);  
  
    // R = 25  
    // sin 30 = 0,5  
    // cos 30 = 0,87  
    // R * sin 30 = 25 * 0,5 = 12  
    // R * cos 30 = 25 * 0,87 = 22  
  
    MoveToEx(hdc, x + 25, y + 25, NULL);  
    LineTo(hdc, x + 25 - 22, y + 25 + 12);  
  
    MoveToEx(hdc, x + 25, y + 25, NULL);  
    LineTo(hdc, x + 25 + 22, y + 25 + 12);  
}
```



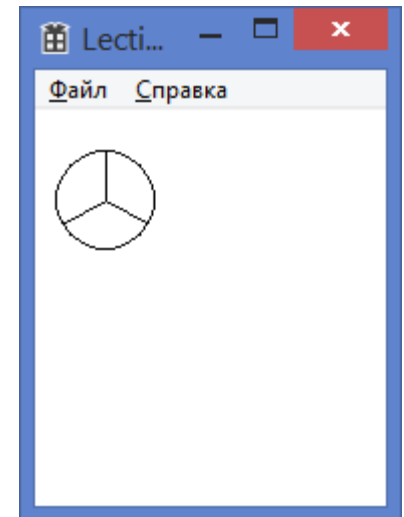
Тригонометрия и Мерседес-Бенц



Рисуем знак зодиака Мерседес-Бенц

```
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam,
LPARAM lParam)
{
    switch (message)
    {
        ...
        case WM_PAINT:
        {
            PAINTSTRUCT ps;
            HDC hdc = BeginPaint(hWnd, &ps);

            MercedesBenz(hdc, 10, 20);
            EndPaint(hWnd, &ps);
        }
        break;
        ...
    }
}
```



Рисуем ряд знаков Мерседес-Бенц

...

```
int x = 10;
```

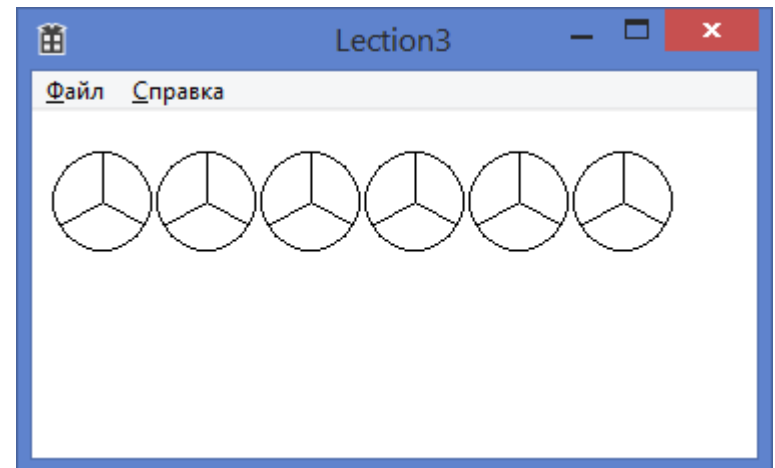
```
do {
```

```
    MercedesBenz(hdc, x, 20);
```

```
    x += 52;
```

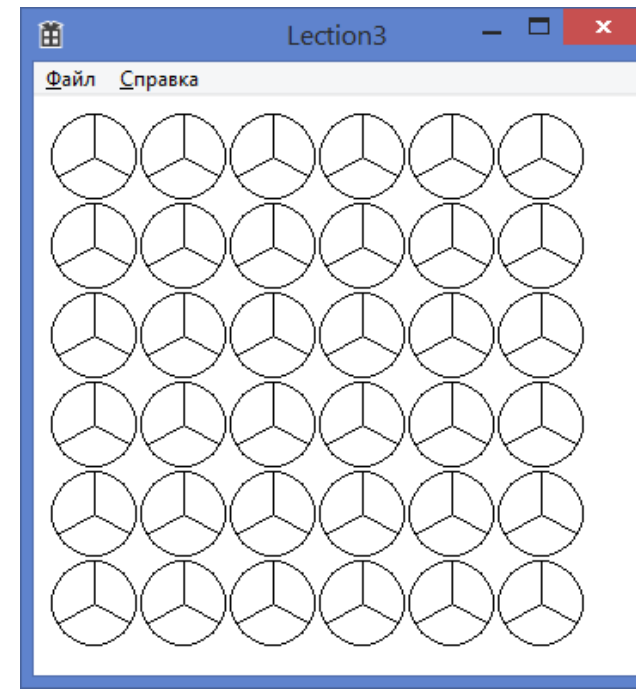
```
} while (x <= 280);
```

...

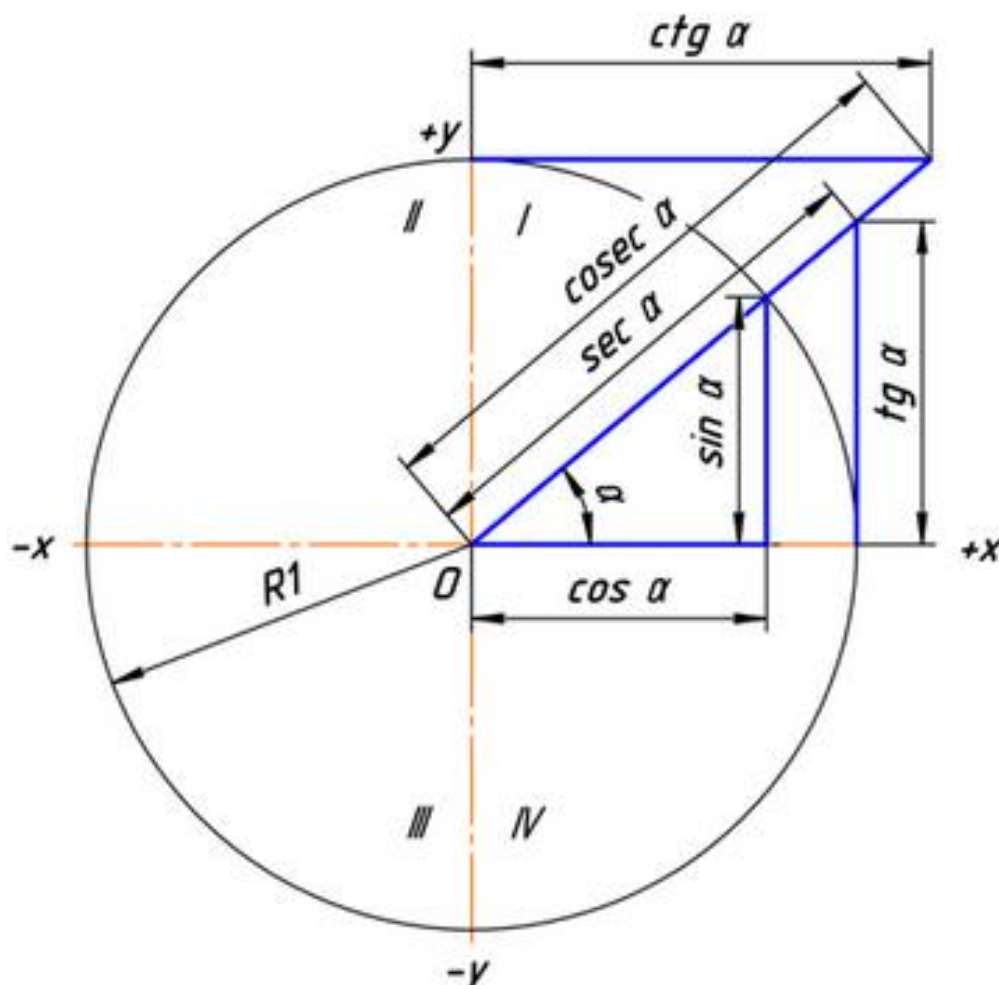


Рисуем поле знаков Мерседес-Бенц

```
int y = 10;  
do {  
  
    int x = 10;  
    do {  
        MercedesBenz(hdc, x, y);  
        x += 52;  
    } while (x <= 280);  
  
    y += 52;  
} while (y <= 270);
```



Вспоминаем тригонометрию



Численные значения тригонометрических функций угла в тригонометрической окружности с радиусом, равным единице

Рисуем окружность из знаков Мерседес-Бенц

...

```
#define _USE_MATH_DEFINES
```

```
#include <math.h>
```

...

```
double alpha = 0;
```

```
int R = 100;
```

```
int cx = 120;
```

```
int cy = 100;
```

```
do {
```

```
    int x = cx + R * cos(alpha);
```

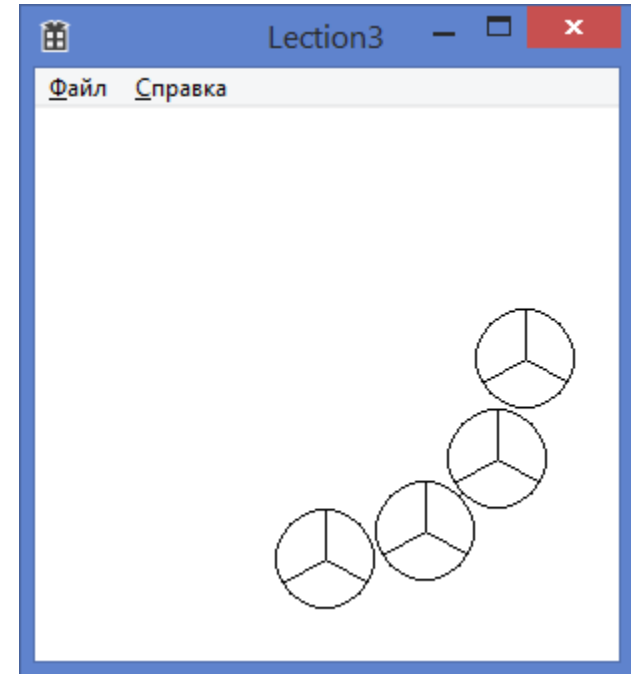
```
    int y = cy + R * sin(alpha);
```

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

```
    alpha += M_PI / 6;
```

```
} while (alpha <= M_PI / 2);
```

...



Рисуем окружность из знаков Мерседес-Бенц

...

```
#define _USE_MATH_DEFINES
```

```
#include <math.h>
```

...

```
double alpha = 0;
```

```
int R = 100;
```

```
int cx = 120;
```

```
int cy = 100;
```

```
do {
```

```
    int x = cx + R * cos(alpha);
```

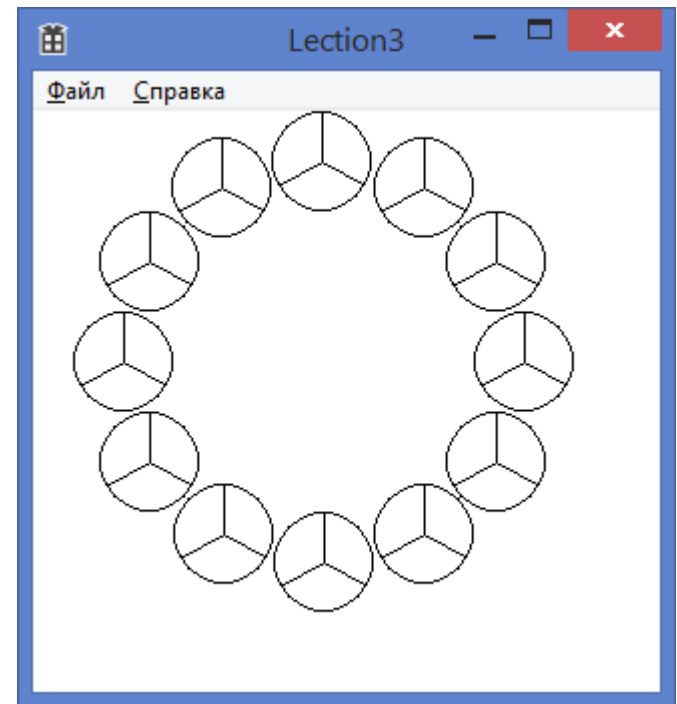
```
    int y = cy + R * sin(alpha);
```

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

```
    alpha += M_PI / 6;
```

```
} while (alpha <= 2 * M_PI);
```

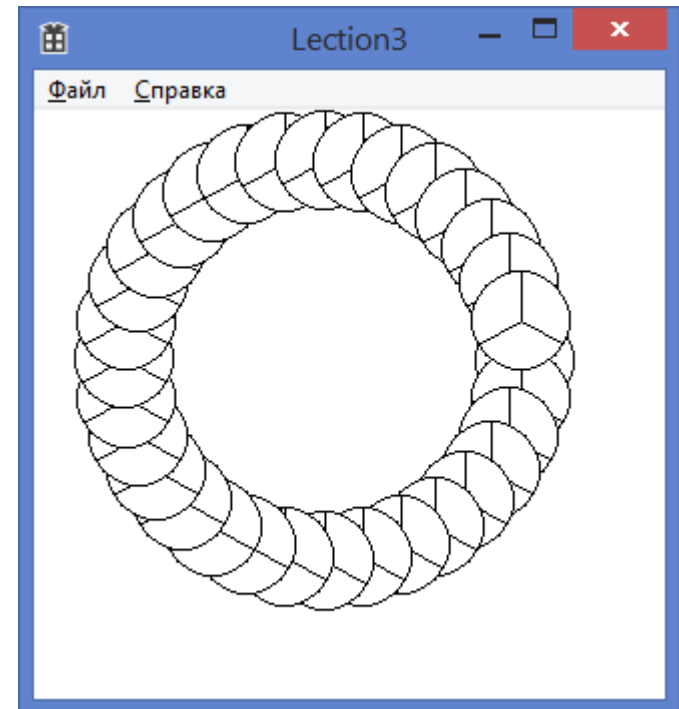
...



Рисуем окружность из знаков Мерседес-Бенц

```
...
#define _USE_MATH_DEFINES
#include <math.h>
...
double alpha = 0;
int R = 100;
int cx = 120;
int cy = 100;

do {
    int x = cx + R * cos(alpha);
    int y = cy + R * sin(alpha);
    MercedesBenz(hdc, x, y);
    alpha += M_PI / 16;
} while (alpha <= 2 * M_PI);
...
```



Рисуем спираль из знаков Мерседес-Бенц

...

```
double alpha = 0;
```

```
int R = 0;
```

```
int cx = 120;
```

```
int cy = 100;
```

```
do {
```

```
    int x = cx + R * cos(alpha);
```

```
    int y = cy + R * sin(alpha);
```

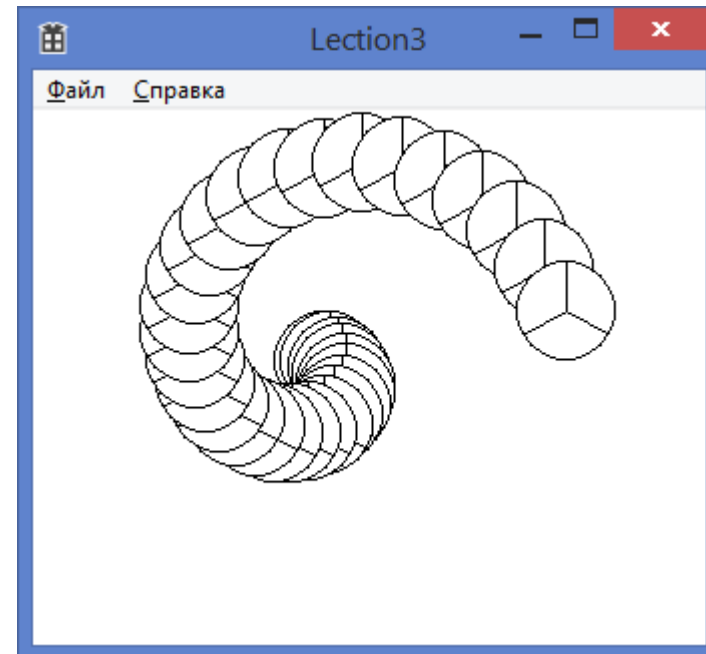
```
    MercedesBenz(hdc, x, y);
```

```
    alpha += M_PI / 16;
```

```
    R += 4;
```

```
} while (alpha <= 2 * M_PI);
```

...



Рисуем спираль из знаков Мерседес-Бенц

...

```
double alpha = 0;
```

```
int R = 0;
```

```
int cx = 120;
```

```
int cy = 100;
```

```
do {
```

```
    int x = cx + R * cos(alpha);
```

```
    int y = cy + R * sin(alpha);
```

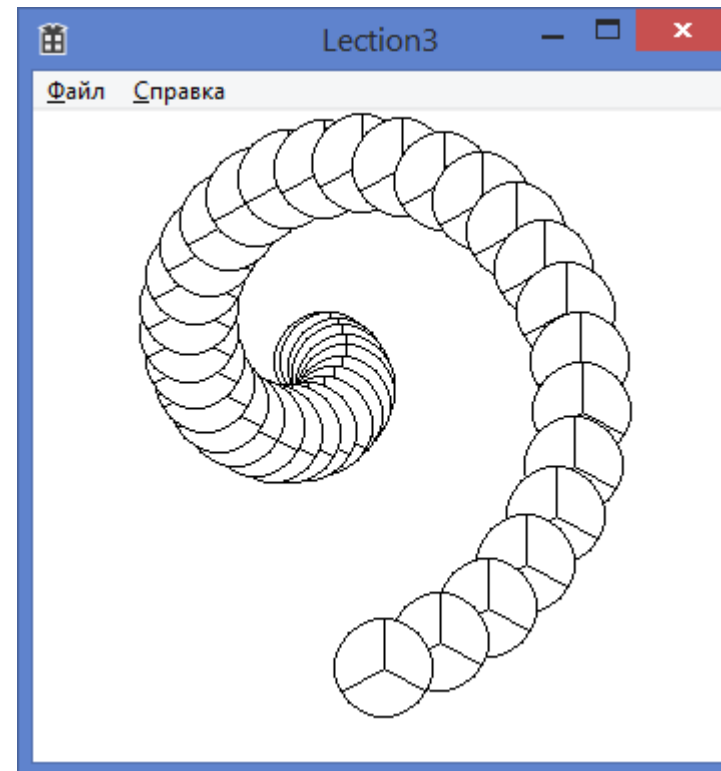
```
    MercedesBenz(hdc, x, y);
```

```
    alpha += M_PI / 16;
```

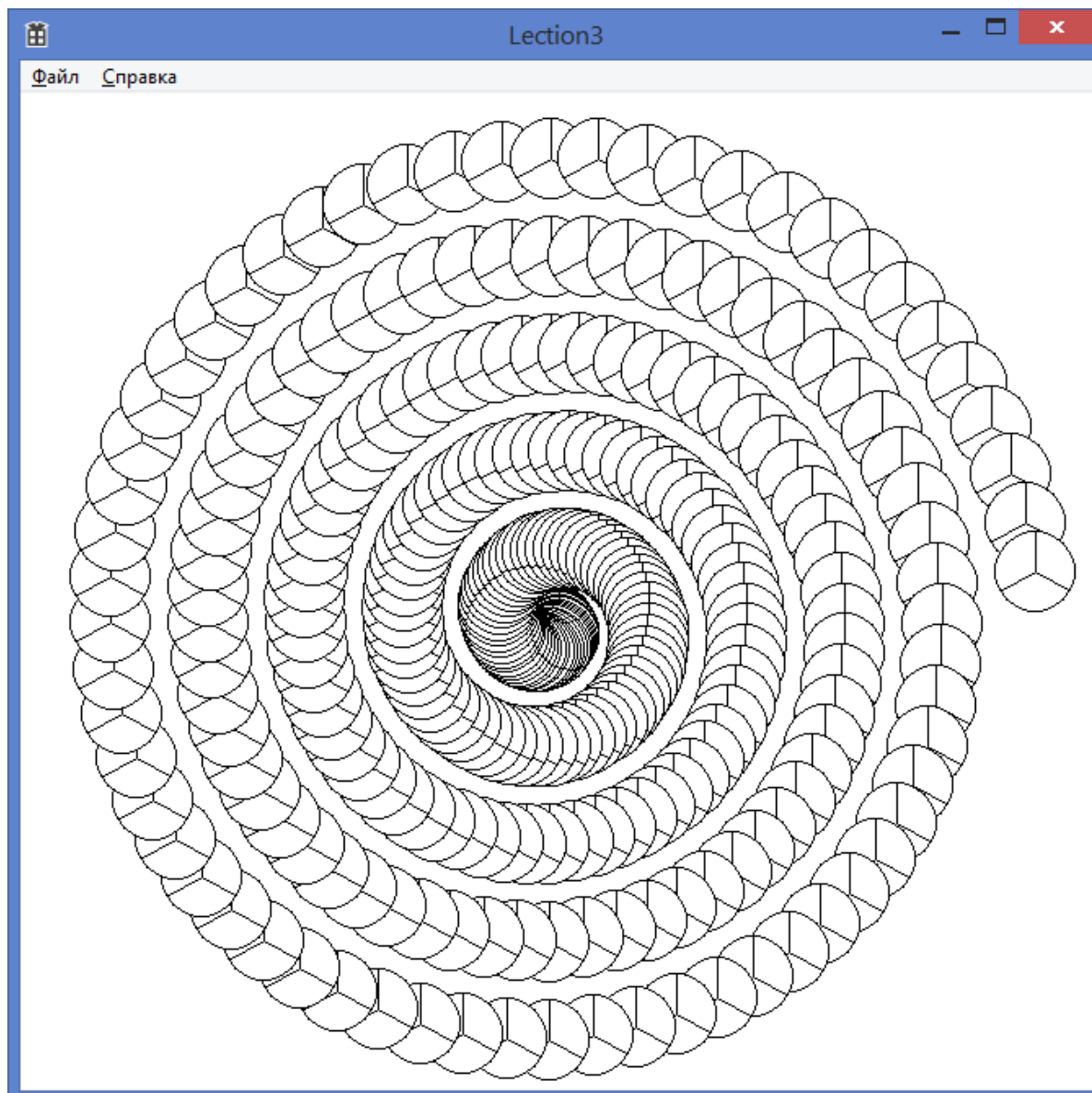
```
    R += 4;
```

```
} while (alpha <= 2.5 * M_PI);
```

...



Рисуем спираль из знаков Мерседес-Бенц



Домашнее задание

1. Нарисовать логотип любимой марки автомобиля (не Мерседес!)
2. Создать функцию для отрисовки логотипа любимой марки автомобиля
3. Логотипами автомобиля нарисовать спираль (по аналогии с лекцией и знаком мерседеса)

Источники информации

- **КАК рисовать в Win32 API? -**

<http://radiofront.narod.ru/htm/prog/htm/winda/api/paint.html>