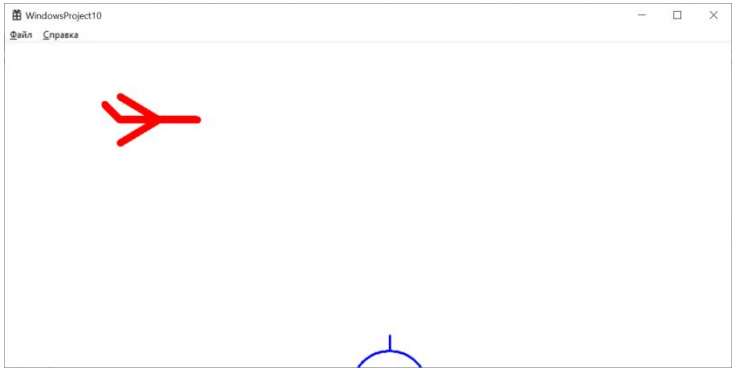
Основы программирования Лабораторная работа №13

WinAPI – Игра
№1
Таймер.
Цикл **while**

Власенко Олег Федосович

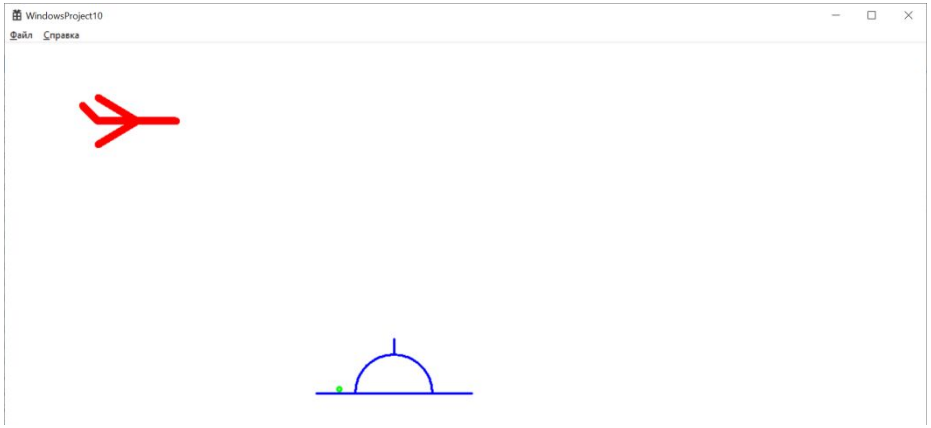
А если окно не того размерчика?

```
BOOL InitInstance(HINSTANCE hInstance, int nCmdShow)
{
   hInst = hInstance; // Сохранить маркер экземпляра в глобальной переменной

   HWND hWnd = CreateWindowW(szWindowClass, szTitle, WS_OVERLAPPEDWINDOW,
      CW_USEDEFAULT, 0, CW_USEDEFAULT, 0, nullptr, nullptr, hInstance, nullptr);

   if (!hWnd)
   {
      return FALSE;
   }

   ShowWindow(hWnd, nCmdShow);
   UpdateWindow(hWnd);

   return TRUE;
}
```

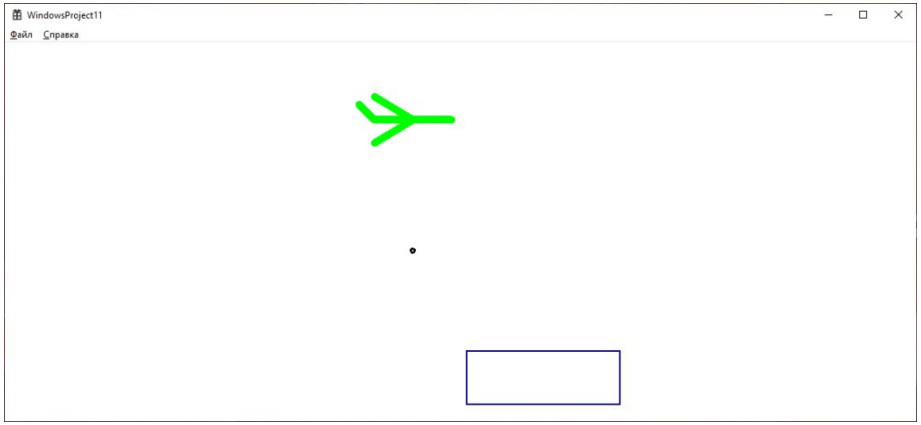


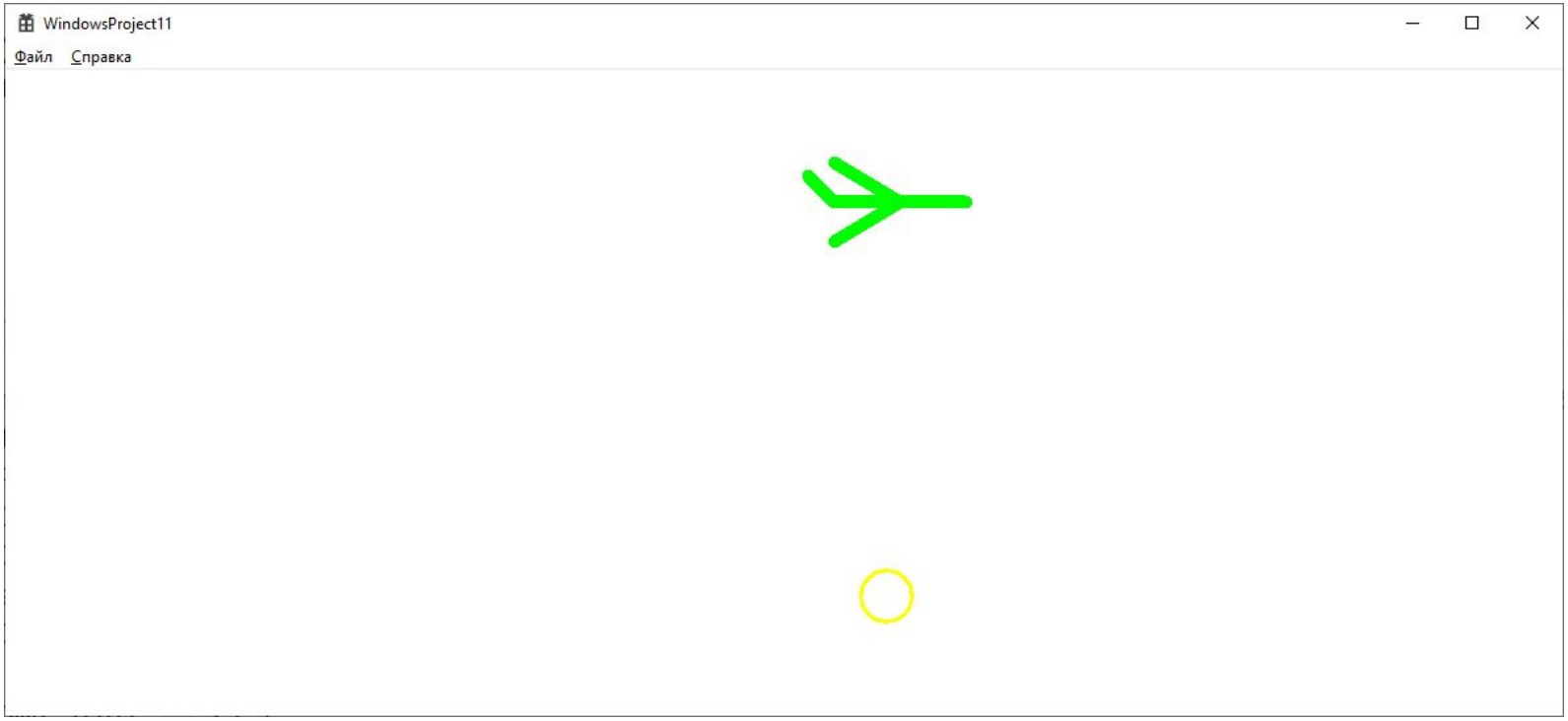
А если окно не того размерчика?

```
BOOL InitInstance(HINSTANCE hInstance, int nCmdShow)
{
   hInst = hInstance; // Сохранить маркер экземпляра в глобальной переменной

   HWND hWnd = CreateWindowW(szWindowClass, szTitle, WS_OVERLAPPEDWINDOW,
      CW_USEDEFAULT, 0, 1200, 550, nullptr, nullptr, hInstance, nullptr);

   if (!hWnd)
   {
      return FALSE;
   }

   ShowWindow(hWnd, nCmdShow);
   UpdateWindow(hWnd);

   return TRUE;
}
```

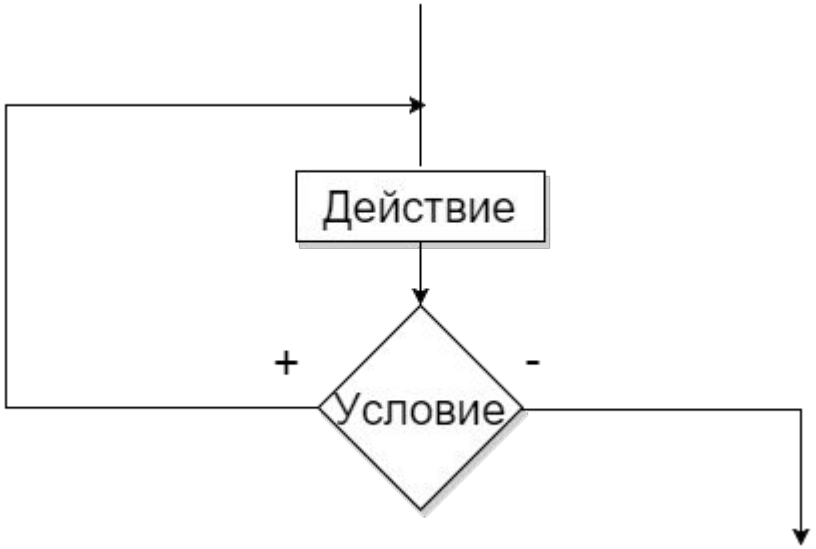


Простейшая игра – разбомби сарай!





Цикл с постусловием do while



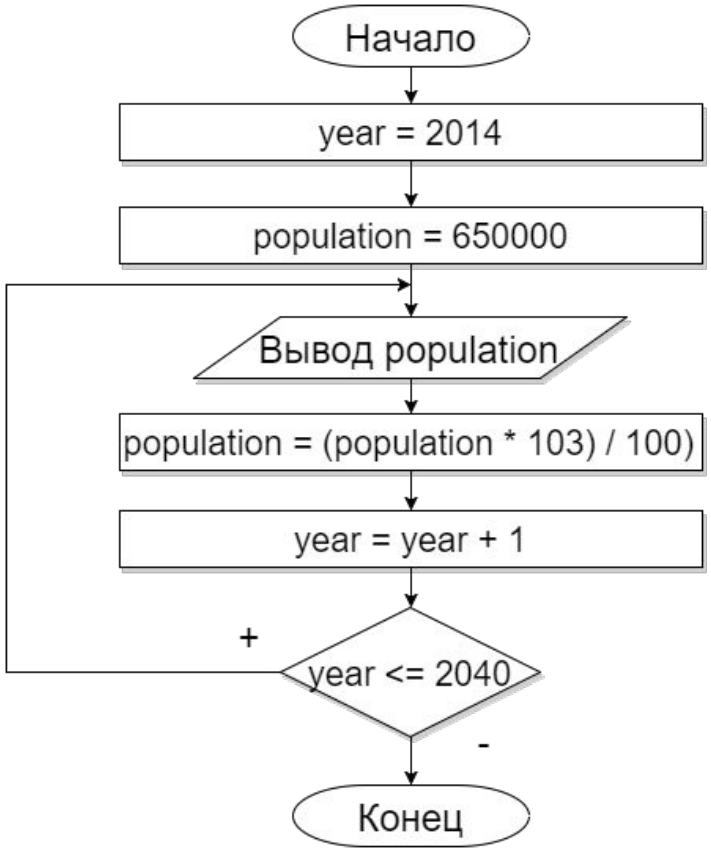

```
do {
    Действие;
} while (Условие);
```

Пример для цикла do while

Население города увеличивается на 3% каждый год. В 2014 году население города составляло 650 000 человек. Напишите программу, которая выведет на экран предсказываемую численность населения города в каждом году, вплоть до 2040.

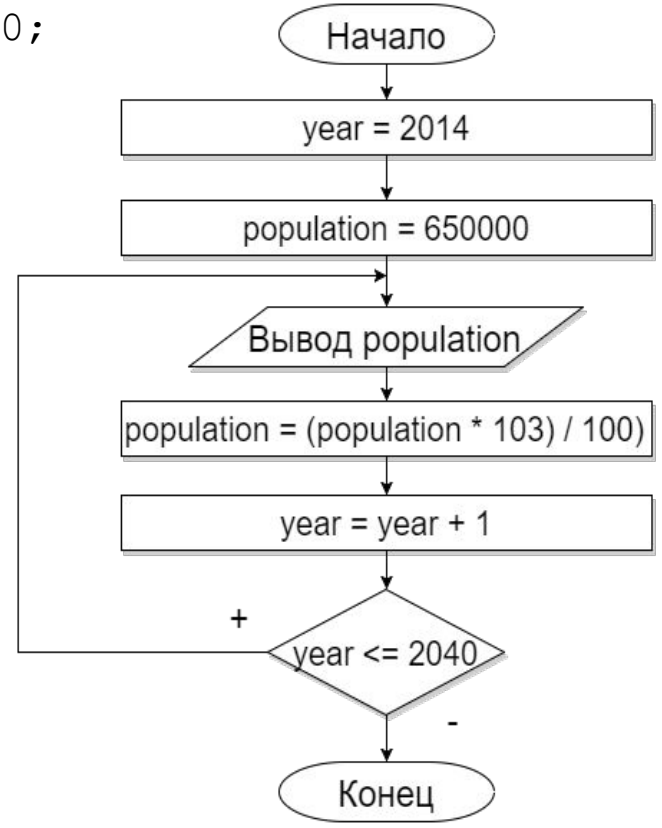
Пример для цикла do while

Население города увеличивается на 3% каждый год. В 2014 году население города составляло 650 000 человек. Напишите программу, которая выведет на экран предсказываемую численность населения города в каждом году, вплоть до 2040.



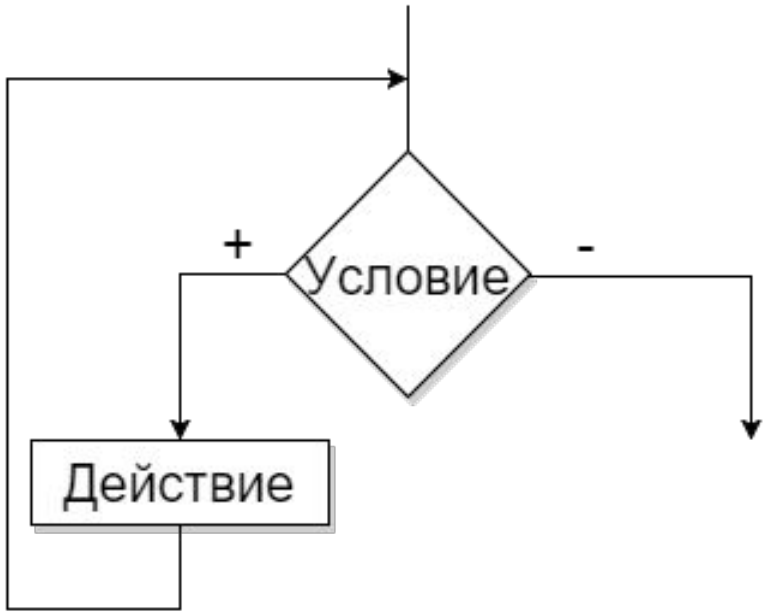
Программа и блок-схема

```
void main() {
    int year = 2014;
    long population = 650000;
    do {
        printf("%li  inhabitants live in the city in %i\n",
               population, year);
        population = (population * 103) / 100;
        year = year + 1;
    } while (year <= 2040);
}
```



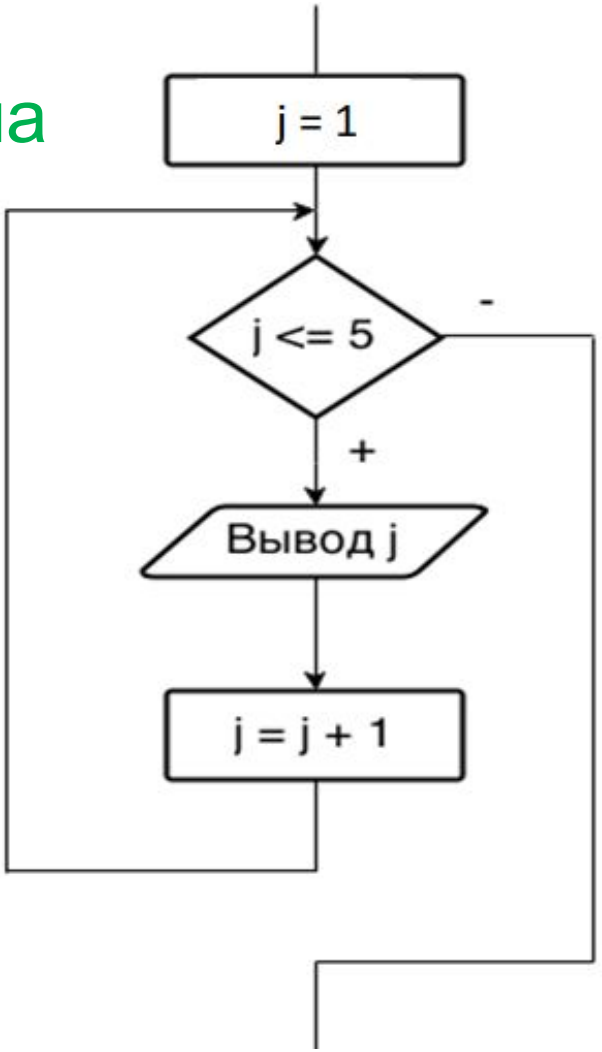
Цикл с предусловием while

```
while (Условие) {
    Действие;
}
```



Пример кода с while

```
int j = 1; // инициализация счетчика цикла
while (j <= 5) { // условие продолжения цикла
    printf("%d ", j);
    j++; // изменение счетчика цикла
}
```



Глобальные переменные

```
// Глобальные переменные:
HINSTANCE hInst;                        // текущий экземпляр
WCHAR szTitle[MAX_LOADSTRING];          // Текст строки заголовка
WCHAR szWindowClass[MAX_LOADSTRING];    // имя класса главного
окна

// самолет
int plane_x = 100;
int plane_y = 100;
int plane_vx = 10;
int plane_vy = 0;
```

Глобальные переменные

```
// бомба
int bomb_state = 0; // 0 - в самолете готова к сбросу, 1 - летит к цели,
                    // 2-9 - взрывается прямо сейчас, 10 - взорвалась
int bomb_x;
int bomb_y;
int bomb_vx;
int bomb_vy;

// цель
int target_exist = 1; // 1 - цель существует, 0 - не существует
int target_x = 600;
int target_y = 400;
int target_width = 200;
int target_height = 70;
```

Рисование самолета

```
void DrawPlane(HDC hdc) {
	HPEN hPen = CreatePen(PS_SOLID, 10, RGB(0, 255, 0));
	SelectObject(hdc, hPen);

	MoveToEx(hdc, plane_x + 50, plane_y, NULL);
	LineTo(hdc, plane_x - 50, plane_y);
	LineTo(hdc, plane_x - 70, plane_y - 20);

	MoveToEx(hdc, plane_x - 50, plane_y + 30, NULL);
	LineTo(hdc, plane_x, plane_y);
	LineTo(hdc, plane_x - 50, plane_y - 30);

	DeleteObject(hPen);
}
```

Перемещение самолета

```
void MovePlane() {

	if (plane_x >= 1000) {
		plane_x = 10;
	}

	plane_x += plane_vx;
	plane_y += plane_vy;
}
```

Сброс бомбы

```
void Bombing() {

	if (bomb_state == 0) {
		bomb_x = plane_x;
		bomb_y = plane_y;
		bomb_vx = plane_vx;
		bomb_vy = 10;
		bomb_state = 1; // бомба летит к цели
	}
}
```

Перемещение (падение) бомбы

```
void MoveBomb() {
	if (bomb_state == 1) {
		bomb_x += bomb_vx;
		bomb_y += bomb_vy;
	}
}
```

Отрисовка бомбы (1)

```
void DrawBomb(HDC hdc) {

    //int bomb_state;
    // 0 - в самолете готова к сбросу, 1 - летит к цели,
    // 2-9 - взрывается прямо сейчас, 10 - взорвалась

    if (bomb_state == 1) {
        HPEN hPen = CreatePen(PS_SOLID, 3, RGB(0, 0, 0));
        SelectObject(hdc, hPen);
        SelectObject(hdc, GetStockObject(NULL_BRUSH));
        Ellipse(hdc, bomb_x - 3, bomb_y - 3, bomb_x + 3, bomb_y + 3);
        DeleteObject(hPen);
    }
```

Отрисовка бомбы (2)

```
else if (bomb_state == 2) {
    HPEN hPen = CreatePen(PS_SOLID, 3, RGB(255, 255, 0));
    SelectObject(hdc, hPen);
    HBRUSH hBrush2 = CreateHatchBrush(HS_DIAGCROSS, RGB(255, 255, 128));
    SelectObject(hdc, hBrush2);

    SelectObject(hdc, GetStockObject(NULL_BRUSH));
    Ellipse(hdc, bomb_x - 10, bomb_y - 10, bomb_x + 10, bomb_y + 10);
    DeleteObject(hPen);
    DeleteObject(hBrush2);
    bomb_state = 3;
}
```

Отрисовка бомбы (3)

```
else if (bomb_state == 3) {
    HPEN hPen = CreatePen(PS_SOLID, 3, RGB(255, 255, 0));
    SelectObject(hdc, hPen);
    HBRUSH hBrush2 = CreateHatchBrush(HS_DIAGCROSS, RGB(255, 255, 128));
    SelectObject(hdc, hBrush2);

    SelectObject(hdc, GetStockObject(NULL_BRUSH));
    Ellipse(hdc, bomb_x - 20, bomb_y - 20, bomb_x + 20, bomb_y + 20);
    DeleteObject(hPen);
    DeleteObject(hBrush2);
    bomb_state = 4;
}
```

Отрисовка бомбы (4)

```
 else if (bomb_state == 4) {
      HPEN hPen = CreatePen(PS_SOLID, 3, RGB(255, 255, 0));
      SelectObject(hdc, hPen);
      HBRUSH hBrush2 = CreateHatchBrush(HS_DIAGCROSS, RGB(255, 255, 128));
      SelectObject(hdc, hBrush2);

      SelectObject(hdc, GetStockObject(NULL_BRUSH));
      Ellipse(hdc, bomb_x - 30, bomb_y - 30, bomb_x + 30, bomb_y + 30);
      DeleteObject(hPen);
      DeleteObject(hBrush2);
      bomb_state = 10;
  }

}
```

Проверка контакта бомбы с чем-нибудь

```
void CheckContact() {
   if (bomb_state == 1) {
      // если попали в цель
      if (target_exist
         &&
         (bomb_x >= target_x) && (bomb_x <= target_x + target_width) &&
         (bomb_y >= target_y) && (bomb_y <= target_y + target_height)
         )
      {
         bomb_state = 2; // бомба начала взрываться
         target_exist = 0; // цель больше не существует!
      }
      // если бомба попала в землю
      else if (bomb_y > 500) {
         bomb_state = 2; // бомба начала взрываться
      }
   }
}
```

Отрисовка цели

```
void DrawTarget(HDC hdc) {
    if (target_exist) {
        HPEN hPen = CreatePen(PS_SOLID, 2, RGB(0, 0, 128));
        SelectObject(hdc, hPen);
        Rectangle(hdc, target_x, target_y,
                        target_x + target_width, target_y + target_height);
        DeleteObject(hPen);
    }
}
```

WndProc (1)

```
//
//  ФУНКЦИЯ: WndProc(HWND, UINT, WPARAM, LPARAM)
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam,
LPARAM lParam)
{
  switch (message)
  {
        …
        case WM_KEYDOWN:
              switch (wParam)
              {
              case VK_RETURN:
                    Bombing();
                    InvalidateRect(hWnd, NULL, TRUE);
                    break;
              }
              break;
```

WndProc (2)

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

	DrawPlane(hdc);
	DrawBomb(hdc);
	DrawTarget(hdc);

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

WndProc (3)

```
case WM_CREATE:
        SetTimer(hWnd, 1, 100, 0);
        break;

case WM_TIMER:
        MoveBomb();
        MovePlane();
        CheckContact();
        InvalidateRect(hWnd, NULL, TRUE);
        break;

…
```

Задача 1

Из исходников собрать работающую игру

Задача 2

Сделать взрыв бомбы не из 3 кадров, а из 6-8 кадров

Задача 3

Самолет несет запас бомб – не единственная бомба, а 6-8 бомб

Задача 3 (1)

Самолет несет запас бомб – не единственная бомба, а 6-8 бомб

Бомба не одна. а N BOMB!

```
// бомба
#define N_BOMB 6

int bomb_state[N_BOMB] = { 0 }; // bomb_state[0] = 0, остальные тоже равны 0!!!
                                // 0 - в самолете готова к сбросу,
                                //1 - летит к цели,
                                // 2-9 - взрывается прямо сейчас,
                                //10 - взорвалась
int bomb_x[N_BOMB]; // не инициализированный массив!
int bomb_y[N_BOMB];
int bomb_vx[N_BOMB];
int bomb_vy[N_BOMB];
```

Задача 3 (2)

Самолет несет запас бомб – не единственная бомба, а 6-8 бомб

Сбрасываем очередную бомбу!

```
void Bombing() {
    int i = 0; // надо пройти по всем бомбам, начиная с бомбы [0]
    while (i < N_BOMB) {

        if (bomb_state[i] == 0) { // найдем первую бомбу, которая еще в самолете
            bomb_x[i] = plane_x;
            bomb_y[i] = plane_y;
            bomb_vx[i] = plane_vx;
            bomb_vy[i] = 10;
            bomb_state[i] = 1; // бомба летит к цели
            return; // дальше не ищем! Улетит только одна бомба!
        }
        i++; // отсутствие увеличения счетчика создаст бесконечный цикл!
    }
}
```

Задача 3 (3)

Самолет несет запас бомб – не единственная бомба, а 6-8 бомб

Перемещение не одной, а N_BOMB бомб!

```
void MoveBomb() {
    int i = 0;
    while (i < N_BOMB) {
        if (bomb_state[i] == 1) {
            bomb_x[i] += bomb_vx[i];
            bomb_y[i] += bomb_vy[i];
        }
        i++;
    }
}
```

Задача 3 (4)

Самолет несет запас бомб – не единственная бомба, а 6-8 бомб

Также нужно изменить код в функциях:
void DrawBomb(HDC hdc)
void CheckContact()

Домашнее задание

- Доделать задачи 1-3, которые не успели сделать в классе.
- Сделайте задачу 4.1 И/ИЛИ 4.2.

Задача 4.1 Ведется счет побед и поражений (попаданий и промахов).

Задача 4.2. Что-то на ваш выбор

Принести на следующее занятие:

1. код финальной игры, чтобы иметь возможность её продемонстрировать и развивать дальше
2. блоксхемы алгоритмов обработки массивов – для каждой из функций где используется цикл while, нужна блоксхем.