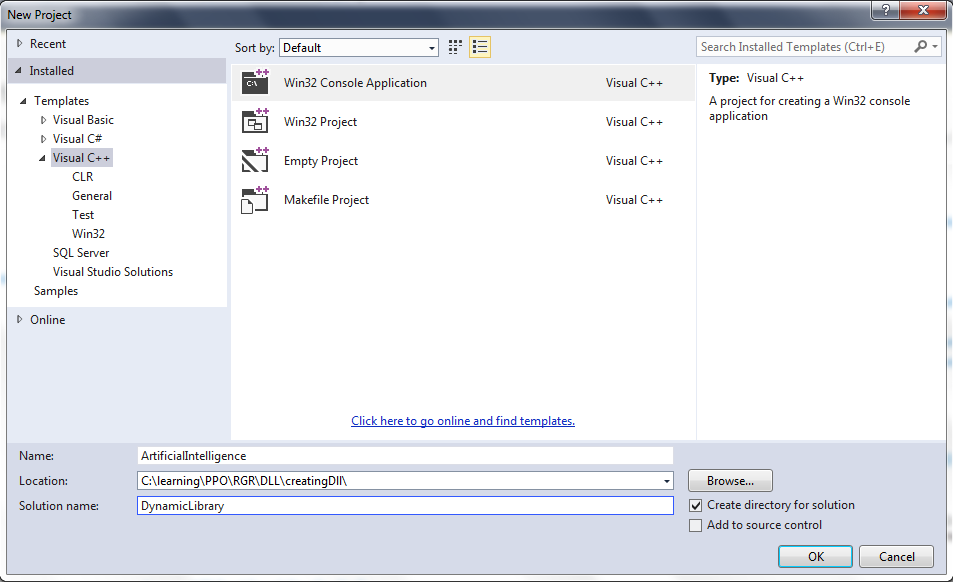
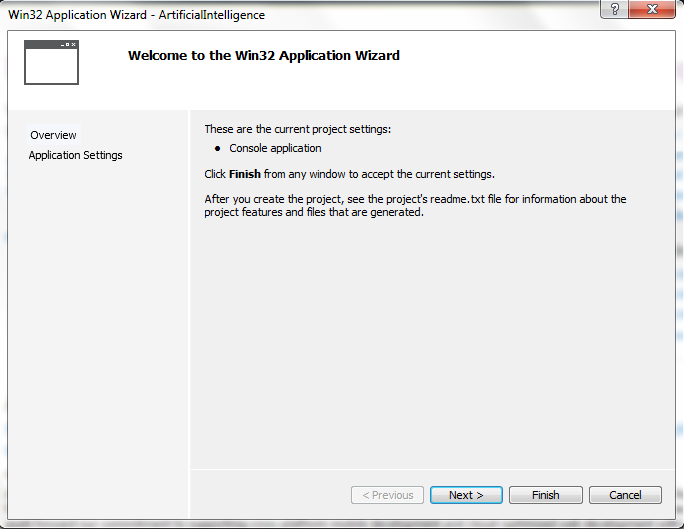
**Создание DLL для управления танком в среде BattleCity:**

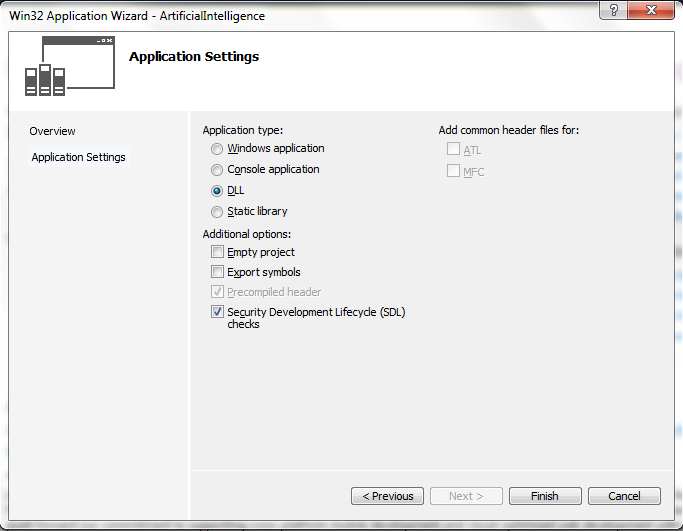
Вообще говоря, создание DLL библиотек было в 6 / 7 лабораторной работе… так есть ли смысл в нашем изложении? Что - то подсказывает, что смысл есть:)

1. **Создаем новый проект в среде Visual Studio (см. Скриншоты)**

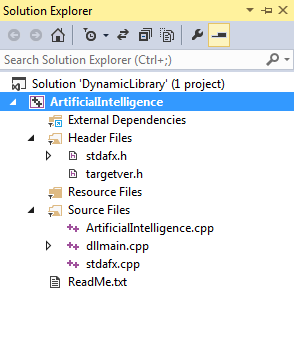




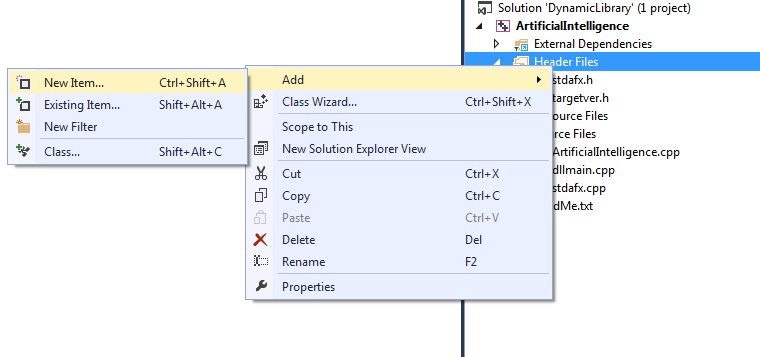
1. **В настройках проекта выбираем подтип DLL (см. Скриншот)**



1. **После проделанных операций solution explorer должен выглядеть следующим образом (см. Скриншот)**



1. **Добавляем новый заголовочный файл ArtificialIntelligence.h**



1. **В ArtificialIntelligence.h добавляем следующее содержимое**

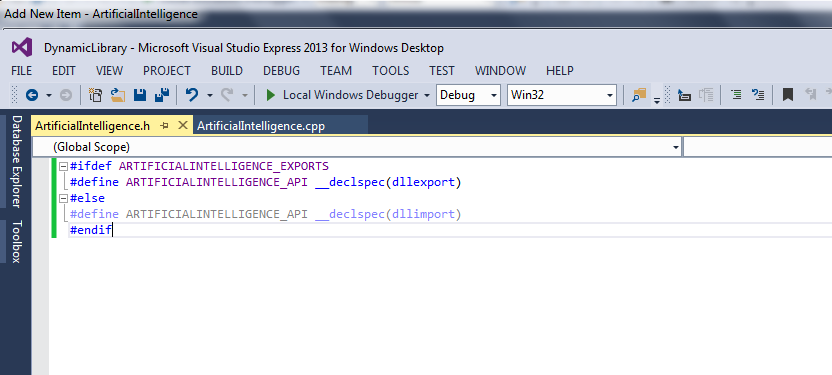
#ifdef ARTIFICIALINTELLIGENCE\_EXPORTS

#define ARTIFICIALINTELLIGENCE\_API \_\_declspec(dllexport)

#else

#define ARTIFICIALINTELLIGENCE\_API \_\_declspec(dllimport)

#endif

Теперь ArtificialIntelligence.h должен у вас выглядеть следующим образом 

1. **Теперь в заголовочный файл (ArtificialIntelligence.h ) добавим функции которые используются для работы DLL в среде Battle City**

extern "C"

{

ARTIFICIALINTELLIGENCE\_API void SetCoords(int x, int y);

ARTIFICIALINTELLIGENCE\_API void SetAngle(int angle);

ARTIFICIALINTELLIGENCE\_API void SetTurretAngle(int angle);

ARTIFICIALINTELLIGENCE\_API void SetCollisionStatus(bool isCollided);

ARTIFICIALINTELLIGENCE\_API void SetLivePercent(int percent);

ARTIFICIALINTELLIGENCE\_API void SetVisilbeEnemyCount(int count);

ARTIFICIALINTELLIGENCE\_API void SetEnemyProteries(int enemyID, int x, int y, int angle, int turretAngle, int livePercent);

ARTIFICIALINTELLIGENCE\_API int GetDirection();

ARTIFICIALINTELLIGENCE\_API int GetRotateDirection();

ARTIFICIALINTELLIGENCE\_API int GetRotateSpeed();

ARTIFICIALINTELLIGENCE\_API int GetTurretRotateDirection();

ARTIFICIALINTELLIGENCE\_API int GetTurretRotateSpeed();

ARTIFICIALINTELLIGENCE\_API int GetFireDistance();

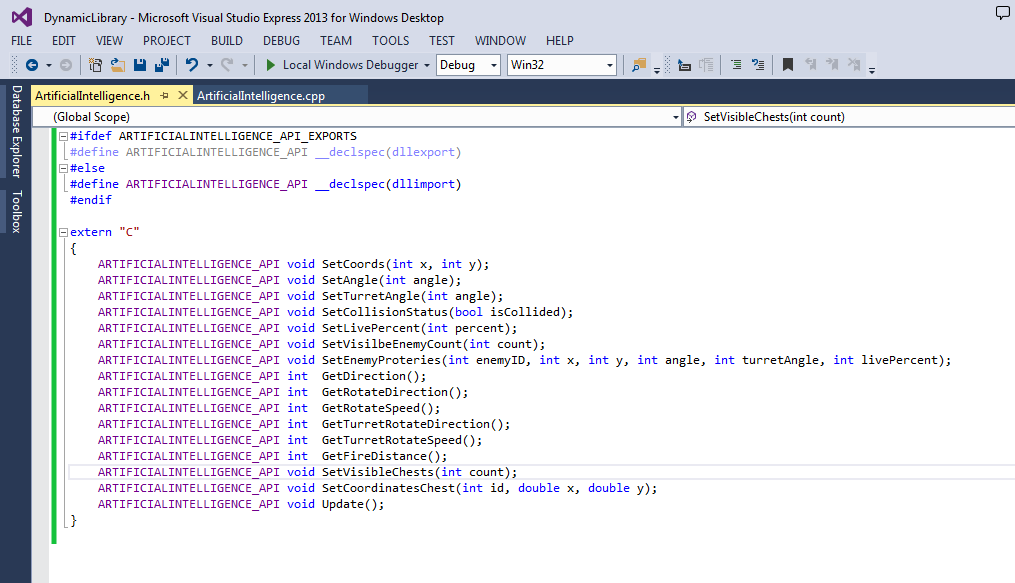
ARTIFICIALINTELLIGENCE\_API void SetVisibleChests(int count);

ARTIFICIALINTELLIGENCE\_API void SetCoordinatesChest(int id, double x, double y);

ARTIFICIALINTELLIGENCE\_API void Update();

}

После данного этапа файл ArtificialIntelligence.h будут выглядеть следующим образом



Поздравляем ваша DLL библиотека успешно создана. Подключайте её к BattleCity и она ...

Не будет работать!!!=) А вы ведь хотите, чтобы она работала? Тогда продолжим!=)

1. **Добавим в файл ArtificialIntelligence.h следующий код (добавляем перед extern “C”):**

struct CTankActions

{

CTankActions() :x(0), y(0), baseAngle(0), turretAngle(0), fireDistance(-1), fireAngle(0) {}

int x;

int y;

int baseAngle;

int turretAngle;

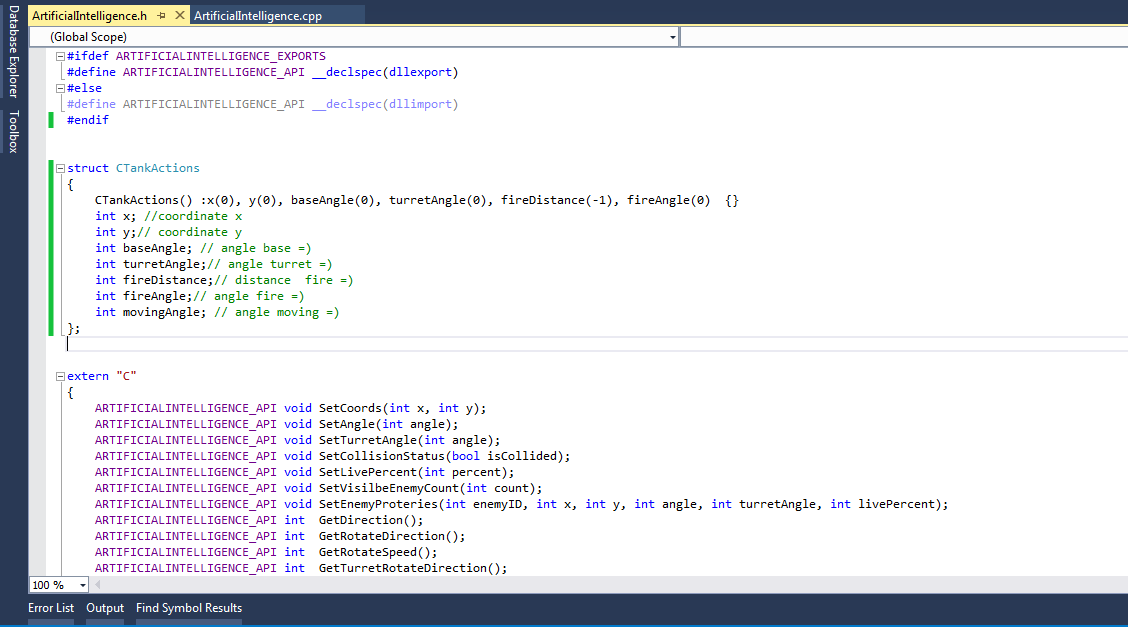
int fireDistance;

int fireAngle;

int movingAngle;

};

И вот скриншот после данного действия



1. **Теперь заполним файл ArtificialIntelligence.cpp**

**Вот этим кодом (см. ниже)**

// AIExample.cpp : Defines the exported functions for the DLL application.

//

#include "stdafx.h"

#include "ArtificialIntelligence.h"

#define \_USE\_MATH\_DEFINES

#include <math.h>

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

CTankActions tank;

int CompareAngles(int a, int b) //сравнение углов

{

while (a < 0)

{

a += 360;

}

while (b < 0)

{

b += 360;

}

int angle = b % 360 - a % 360;

if (angle > 180)

{

angle -= 360;

}

if (angle < -180)

{

angle += 360;

}

return angle;

}

double GetDistance(int x1, int x2, int y1, int y2) // получить расстояние

{

return sqrt((float)((x1 - x2)\*(x1 - x2) + (y1 - y2)\*(y1 - y2)));

}

extern "C"

{

ARTIFICIALINTELLIGENCE\_API void SetCoords(int x, int y) //устанавливаем координаты танка переданные в DLL

{

tank.x = x;

tank.y = y;

}

ARTIFICIALINTELLIGENCE\_API void SetAngle(int angle) //устанавливаем угол поворота базы танка переданные в DLL

{

tank.baseAngle = angle;

}

ARTIFICIALINTELLIGENCE\_API void SetTurretAngle(int angle) //устанавливаем угол поворота башни танка

{

tank.turretAngle = angle;

}

ARTIFICIALINTELLIGENCE\_API void SetCollisionStatus(bool isCollided) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API void SetLivePercent(int percent) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API void SetVisilbeEnemyCount(int count) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API void SetEnemyProteries(int enemyID, int x, int y, int angle, int turretAngle, int livePercent) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API int GetDirection() //возвращаем направление движения танка

{

return 0;

}

ARTIFICIALINTELLIGENCE\_API int GetRotateDirection() //возвращаем направление поворота танка

{

if (CompareAngles(tank.baseAngle, tank.movingAngle) > 0)

{

return 1;

}

if (CompareAngles(tank.baseAngle, tank.movingAngle) < 0)

{

return -1;

}

return 1;

}

ARTIFICIALINTELLIGENCE\_API int GetRotateSpeed() //скорость поворота танка

{

return 10;

}

ARTIFICIALINTELLIGENCE\_API int GetTurretRotateDirection() //возвращаем направление поворота башни танка

{

if (CompareAngles(tank.turretAngle, tank.fireAngle) > 0)

{

return 1;

}

if (CompareAngles(tank.turretAngle, tank.fireAngle) < 0)

{

return -1;

}

return 0;

}

ARTIFICIALINTELLIGENCE\_API int GetTurretRotateSpeed() //скорость поворота башни танка

{

if (abs(CompareAngles(tank.turretAngle, tank.fireAngle)) < 20)

{

return abs(CompareAngles(tank.turretAngle, tank.fireAngle));

}

return 20;

}

ARTIFICIALINTELLIGENCE\_API int GetFireDistance() //дальность выстрела (120 - значение по умолчанию)

{

return 120;

}

ARTIFICIALINTELLIGENCE\_API void SetVisibleChests(int count) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API void SetCoordinatesChest(int id, double x, double y) //не используется в данной DLL

{

}

ARTIFICIALINTELLIGENCE\_API void Update()

{

tank.movingAngle = tank.baseAngle; //угол движения тот же что и у положения танка

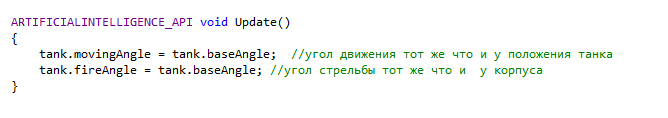
tank.fireAngle = tank.baseAngle; //угол стрельбы тот же что и у корпуса

}

}

P.S. Полное описание функций находится в «Документации по BattleCity.Net»

Функция Update():



1. **После компиляции под режимом Release зайдите в папку с проектом (далее Release), копируйте dll библиотеку и вставляйте ее в папку с приложением Battle City.NET**
2. **Запускайте игру и наслаждайтесь:D**



**P. S.** Мы рассмотрели простейший пример искусственного интеллекта. Более сложные примеры рассмотрены у нас в других интеллектах, но вы кончено же можете улучшить их или создать какой-нибудь свой неповторимый интеллект. Дерзайте!