МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики и технологий программирования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

«Консольный тетрис в Visual studio C++»

Выполнил:

студент гр.653502

Цыганов А.А.

Проверил: Козуб В.Н.

г. Минск

2017 год

**СОДЕРЖАНИЕ**

**1. ВВЕДЕНИЕ**

**2. ЦЕЛИ И ЗАДАЧИ**

**3. ПРАВИЛА ИГРЫ**

**4. РАЗРАБОТКА ПРОГРАММЫ**

**5. ЗАКЛЮЧЕНИЕ**

**6. СПИСОК ИСТОЧНИКОВ**

**Введение.**

Игры всегда были интересны людям. Трудно сказать, когда появились первые игры, но совершенно ясно, что они являются неотъемлемой частью жизни человека. Уже в средние века игры были популярны как среди знатных людей, так и среди простого народа. Приводя примеры популярных древних интеллектуальных игр, можно упомянуть шахматы, домино и игру Го.

Мой проект является консольной реализацией одной из наиболее простых и известных игр - Тетрис.

Игра реализована практически на всех современных компьютерах, включая КПК, мобильные телефоны, игровые видеоприставки, телевизоры (как доп. функция), множество карманных игровых устройств. Есть варианты игры для всех сколько-нибудь распространённых ОС, а также для Java. Есть порт даже для осциллографа. Трудно, если вообще возможно, назвать такую вычислительную платформу, где бы не было этой игры.

Для написания кода я использовал Visual Studio C++. Для курсового проекта был выбран язык программирования C++, так как он предназначен для разработки самых разнообразных приложений, но при этом сохраняет выразительность и элегантность, присущую языкам C.

Целями курсового проекта являются:

• Закрепление знаний среды разработки С++.

• Изучение графических возможностей консоли.

**Цели и задачи.**

Основной целью данной курсовой работы является изучение и создание игры тетрис с использованием консоли, а также программирование компонентов на языке C++.

Данная работа состоит из 3 разделов: первый раздел теоретический, содержащий общие сведения по теме, второй раздел практический, где описывается код и идея игры. Третий раздел – руководство пользователя.

Основные задачи, выполнение для написания работы:

* Изучение методов для создания графических объектов в консоле.
* Изучение жанра игр-головоломок.
* Изучение правил тетриса.
* Написание кода программы
* Создание игры жанра головоломка.
* Анализ полученной игры

**Правила игры.**

Случайные фигурки тетрамино падают сверху в прямоугольный стакан. В полёте игрок может поворачивать фигурку на 90° и двигать её по горизонтали. Также можно «сбрасывать» фигурку, то есть ускорять её падение, когда уже решено, куда фигурка должна упасть. Фигурка летит до тех пор, пока не наткнётся на другую фигурку либо на дно стакана. Если при этом заполнился горизонтальный, он пропадает и всё, что выше него, опускается на одну клетку. Дополнительно показывается фигурка, которая будет следовать после текущей — это подсказка, которая позволяет игроку планировать действия. Игра заканчивается, когда новая фигурка не может поместиться в стакан. Игрок получает очки за каждый заполненный ряд, поэтому его задача — заполнять ряды, не заполняя сам стакан (по вертикали) как можно дольше, чтобы таким образом получить как можно больше очков.

**Разработка программы.**

Далее будут описаны основные функции, а так же этот раздел будет руководством по созданию программы.

Возможно 2 режима игры для выбора:

1) Стакан с начала игры пуст:



2) Стакан случайно заполнен фигурами или их частями:



1) Как и в любом тетрисе присутствует 7 видов фигур, описанных в виде матриц 4х4 вида:

int fmap[FMAP\_COUNTS][4][4] =

{

{

{ 1, 1, 0, 0 }, // \* \*

{ 1, 1, 0, 0 }, // \* \*

{ 0, 0, 0, 0 },

{ 0, 0, 0, 0 }

},

{

{ 1, 0, 0, 0 }, // \*

{ 1, 0, 0, 0 }, // \*

{ 1, 0, 0, 0 }, // \*

{ 1, 0, 0, 0 } // \*

},

{

{ 0, 0, 1, 0 }, // \*

{ 1, 1, 1, 0 }, // \* \* \*

{ 0, 0, 0, 0 },

{ 0, 0, 0, 0 }

},

{

{ 1, 1, 1, 0 }, // \* \* \*

{ 0, 0, 1, 0 }, // \*

{ 0, 0, 0, 0 },

{ 0, 0, 0, 0 }

},

{

{ 0, 1, 1, 0 }, // \* \*

{ 1, 1, 0, 0 }, // \* \*

{ 0, 0, 0, 0 },

{ 0, 0, 0, 0 }

},

{

{ 1, 1, 0, 0 }, // \* \*

{ 0, 1, 1, 0 }, // \* \*

{ 0, 0, 0, 0 },

{ 0, 0, 0, 0 }

},

{

{ 1, 0, 0, 0 }, // \*

{ 1, 1, 0, 0 }, // \* \*

{ 1, 0, 0, 0 }, // \*

{ 0, 0, 0, 0 }

}

};

Элементы управления:

• Перемещение фигур осуществляется с помощью стрелок вправо, влево.



Вызов поворота:

case KEY\_LEFT:

if (valnewpos(px - 1, py)) px--;

break;

case KEY\_RIGHT:

if (valnewpos(px + 1, py)) px++;

break;

Функция valnewpos:

int valnewpos(int x, int y)

{

int i, j;

if (x < 0) return FALSE;

for (i = 0; i < 4; i++)

{

for (j = 0; j < 4; j++)

{

if (map[j][i])

{

if ((j + x >= SIZEX) || (i + y >= SIZEY)) return FALSE;

if (screen[j + x][i + y])

{

return FALSE;

}

}

}

}

return TRUE;

}

• Поворот фигуры осуществляется с помощью стрелки вверх или пробела.

 

Функция поворота:

#define inv(x) ((x \* (-1)) + 3) // inv(1) = 2

void rotatemap(void)

{

int \_map[4][4];

int i, j, sx = 4, sy = 4;

for (i = 0; i < 4; i++)

for (j = 0; j < 4; j++)

{

\_map[j][i] = map[j][i];

if (map[j][i])

{

if (i < sx) sx = i;

if (inv(j) < sy) sy = inv(j);

}

map[j][i] = 0;

}

for (i = 0; i < 4; i++)

for (j = 0; j < 4; j++)

if (\_map[inv(i)][j]) map[j - sx][i - sy] = 1;

if (!valnewpos(px, py))

for (i = 0; i < 4; i++)

for (j = 0; j < 4; j++)

map[j][i] = \_map[j][i];

}

• Присутствует возможность поставить игру на паузу нажав P.



Пауза:

case 'p': // pause

\_getch();

break;

Функция удаления строки:

void deleteline(void)

{

int i, j, k, cl;

for (i = SIZEY - 1; i >= 0; i--)

{

cl = 1;

for (j = 0, cl = 1; j < SIZEX; j++)

{

if (!screen[j][i]) cl = 0;

}

if (cl)

{

gotoxy(0, i);

for (k = 0; k < SIZEX; k++) putchar('\_'), sleep(20); // anim

score += (((i \* (-1)) + SIZEY) \* 10);

for (k = i; k > 0; k--)

{

for (j = 0; j < SIZEX; j++)

{

screen[j][k] = screen[j][k - 1];

}

}

i++;

print();

}

}

}

Простейшее game menu:



Информация об управлении и авторе в разделе Info:



**Заключение.**

В результате работы над данным курсовым проектом была разработана игровая программа Тетрис и документация к ней. В процессе работы над программой были развиты навыки работы с компонентами среды Visual Studio, языка программирования С++.

**Список источников:**

[http://www.cyberforum.ru](http://www.cyberforum.ru/)

h[ttps://en.wikipedia.org](https://en.wikipedia.org/)

[https://habrahabr.ru](https://habrahabr.ru/)

[http://www.cplusplus.com](http://www.cplusplus.com/)

<https://msdn.microsoft.com/en-us/library/60k1461a.aspx>

[http://www.programmersforum.ru](http://www.programmersforum.ru/)