

## ЛИТЕРАТУРА

1. Краткий курс теории обработки изображений [Электронный ресурс] – Режим доступа: <https://hub.exponenta.ru/post/kratkiy-kurs-teorii-obrabotki-izobrazheniy734> (дата обращения 23.04.2021)

2. OpenCV-PythonTutorials'sdocumentation [Электронный ресурс] – Режим доступа: <https://opencv-python-tutroals.readthedocs.io/> (дата обращения 23.04.2021)

УДК 004.9: 004.4

Студ. Р.Л. Вайсера, М.С. Левша  
Науч. рук. ст. преп. А.С. Наркевич  
(кафедра программной инженерии, БГТУ)

### ИГРА «EXCELLENCE»

Платформа Unity – это мультиплатформенный инструмент для разработки двух- и трехмерных приложений и игр, работающий под операционными системами Windows и OS X, Android, Apple iOS, Linux, а также на игровых приставках.

Игра «Excellence» представляет собой аркаду в космической тематике. Цель игры «Excellence» – набрать как можно больше очков, уничтожая астероиды, летящие навстречу игроку, и корабли противника. Так же имеется рекордный счет, который обновляется, если игрок набрал большее количество очков. Игрок управляет космическим истребителем, который может двигаться влево, вправо, вверх и вниз. При переходе на следующий уровень, сила вражеского корабля увеличивается. Игра заканчивается при уничтожении корабля игрока.

При запуске игры пользователь попадает в стартовое меню, в котором находятся три кнопки (Start, Help, Exit). С помощью кнопки Start пользователь попадает в игровое пространство. Кнопка Exit отвечает за закрытие приложения. При нажатии на кнопку Help появляется основная информация об игровом процессе. В момент смерти игрока появляется сцена конца игры. В ней присутствуют кнопки (Retry, Exitto menu). Пользователь может начать игру заново, нажав кнопку Retry или попасть в стартовое меню, нажав кнопку Exitto menu. Игровая сцена состоит из компонентов: фон, игрок, объект создания астероидов, ловушка для астероидов, источники холодного и теплого света.

Теперь поговорим о реализации ракеты. Ракета ищет местонахождение астероида так же, как и бот ищет позицию игрока. Чтобы ракета наводилась на астероид, вычисляется разница между позицией астероида и запущенной ракетой, `Vector.Slerp (gameObject.transform.up,`

`target_vector, 2f)` поворачивает ракету в направлении цели, `game Object.transform.Translate (Vector3.up*0,050f)` перемещает ракету в направлении астероида.

Создание собственной игры – это исключительный опыт, который помог нам освоить азы движка Unity и дал нам толчок для развития в дальнейшем.

УДК 004.588

Студ. В.А. Белашков

Науч. рук. ст. преп. А. С. Наркевич  
(кафедра программной инженерии, БГТУ)

## САМОУЧИТЕЛЬ АНГЛИЙСКОГО ЯЗЫКА

Программа является консольным приложением, написанным на языке программирования C++. Проект состоит из 7 файлов: *main.cpp*, *stdafx.h*, *Leaing.cpp*, *Text.cpp*, *Functions.cpp*, *Dictionary.h*, *Dictionary.cpp*.

Файл *main.cpp* содержит главную функцию, которая вызывает остальные функции. Заголовочный файл проекта *stdafx.h* содержит заголовки используемых библиотек, прототипы функций, определение структур (*Settings* – настройки для изучения, *Grammar* – массивы со словами для изучения грамматики) и глобальных переменных.

*Learning.cpp* содержит функции для блока изучения: *learningWord Call* (нужна для вызова функции *learningWord* и передачи в нее параметров изучения), *learning Word* (изучение слов), *learning Gram* (изучение грамматики), *correctEnding* (т.к. слова для изучения грамматики используются из словаря, их нужно привести к нужной форме или числу, что и делает эта функция).

Файл *Text.cpp* (файл с информационными функциями) содержит все функции *menu* и функцию *grammarAid*. *Functions.cpp* (файл со вспомогательными функциями): *start* (считывание информации из файлов (словарь и настройки) и заполнение ей соответствующих структур), *end* (заполнение файлов (словарь и настройки) из соответствующих структур), *comparisonStr* (сравнение двух строк на равенство, без учета регистра).

Словарь реализован в виде пространства имен (*Dictionary*) и состоит из двух файлов. Первый *Dictionary.h* содержит определение структуры словаря и прототипы функций *add Word* (добавление слова в словарь), *del Word* (удаление слова из словаря), *check Dictionary* (просмотр словаря). *Dictionary.cpp* содержит определение функций, объявленных в *Dictionary.h*.