МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий

Кафедра технологий программирования

**ЛАБОРАТОРНАЯ РАБОТА №12**

по дисциплине: **«Надёжность программного обеспечения»**

на тему: «**Основы модульного тестирования»**

Вариант: игра «Звёздные войны»

ВЫПОЛНИЛ студент группы

Яблонский А.Д.

ПРОВЕРИЛ ст. преподаватель

Данченко Е.В.

Полоцк, 2018 г.

**Цель работы**: ознакомиться с основами модульного, объектно-ориентированного и пошагового тестирования. Научиться принципам тестирования структуры программных модулей.

**Анализ задания, с описанием своего варианта:** Провести модульное тестирование для программного продукта из предыдущей лабораторной работы. Составить отчет о проделанной работе.

**Краткое описание программы из варианта задания**

Приложение «Звёздные войны» представляет собой игровое приложение. Смысл игры заключается в уклонении и уничтожении вражеских кораблей.

# Сводка по результатам тестирования

Отчет обнаружения ошибок представлен в таблице 1.

**Таблица 1** – Ошибки обзора

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ Ошибки** | **Название модуля/функции** | **Описание ошибки** | **Важность**  **ошибки**  **(высокая, средняя, низкая)** | **Ошибка исправлена**  **Да/Нет** |
| 1 | moveTo(); | Во время игры корабль игрока не двигается | Высокая | Да |
| 2 | startGameplayMusic(); | Во время запуска игры не звучит музыка | Низкая | Да |
| 3 | recalculateScore(); | Во время игры счет очков неверный | Высокая | Да |
| 4 | isEnemyHit(); | В момент столкновения с врагом игра не заканчивается | Высокая | Да |

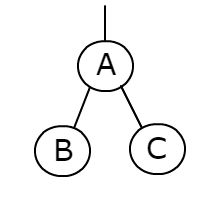
# Графы и вычисление цикломатических чисел

Метрический показатель сложности или цикломатическое число ***g*** потокового графа определяется по формуле:

**G=r-v+2,**

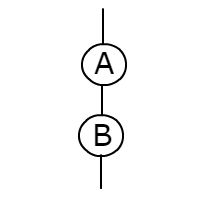
Где ***r*** – количество ребер графа,

***V*** – количество вершин графа.



**Рисунок 1** – цикломатическая схема метода moveTo ();

Метрический показатель сложности или цикломатическое число g=3-3+2 = 2.



**Рисунок 2** – цикломатическая схема метода save();

G=2-2+2 = 2.

**Тестовые сценарии для модульного тестирования**

Пример сценария приведен в таблице 2.

**Таблица 2** – Тестовые сценарии

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | **№ сценария** | **Описание прохода** | **Контрольные примеры, позволяющие реализовать описанную ситуацию** | **Тест пройден**  **Да/Нет** |
| G = 2 | 1 | A-B | State=menu(); | Да |
| 2 | A-C | state=gameplay(); | Да |
| G = 2 | 1 | A-B | gameScore=Записываем количество набранных очков | Да |

# Схема взаимодействия модулей

Main.cpp

Game.h

Sound.h

Player.h

Score.h

Enemy.h

**Рисунок 3** – Модульная схема

Пример взаимодействия модулей приведен в таблице 3.

**Таблица 3** – Взаимодействие модулей

| **№ в последовательности** | **Описание последовательности** | **Контрольные примеры, позволяющие реализовать описанную ситуацию** | **Тест пройден**  **Да/Нет** |
| --- | --- | --- | --- |
| 1 | main -> Game | ID=Start(); | Да |
| 2 | Game -> Sound | ID=soundSound (); | Да |
| 3 | Game -> Enemy | ID=moveEnemy(); | Да |
| 4 | Game -> Score | ID=soundScore(); | Да |
| 5 | Game -> Person | ID=movePlayer(); | Да |

**Вывод:** в ходе выполнения данной лабораторной работы я ознакомился с модульным тестированием на примере игрового приложения «Звёздные войны». Построил схему взаимодействия нескольких модулей. Изучил различные возможные сценарии для модульного тестирования.