

**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ** **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** **«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ** **«СИНЕРГИЯ»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | 09.02.07 Информационные системы и программирование |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

.

**Отчет по лабораторной работе №1**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **на тему** | |  | **Разработка тестового сценария проекта** | | | | |
|  | | | | | |  | (наименование темы) |
|  |  | |  | | | | |
| **по дисциплине** | | | |  | Тестирование информационных систем | | |
|  | | | | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | Никитин Александр Евгеньвич |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | ДКИП-312 |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Авдеенков Владимир Александрович |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2024**

**Практическая работа №1: «Разработка тестового сценария проекта»**

**Цель занятия:** Получить навыки разработки тестовых сценариев.

**Задачи:**

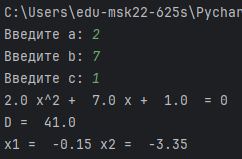
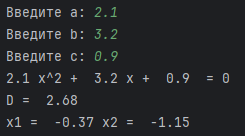
1. Написать (язык – любой) программу решения квадратного уравнения ах2 + bх + с = 0 (задаваемые с клавиатуры коэффициенты – a, b и c);
2. Найти минимальный набор тестов для программы нахождения корней квадратного уравнения ах2 + bх + с = 0 (корни вещественные или комплексные, один/два/бесконечно, один/оба нулевых корня и т. д.);
3. Оформить отчёт (код и скриншоты работы каждого теста, выводы).

**Ход работы:**

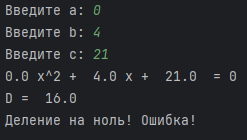
**Задание №1:**

import math  
  
a = float(input("Введите a: "))  
b = float(input("Введите b: "))  
c = float(input("Введите c: "))  
print(a, "x^2 + ", b, "x + ", c, " = 0")  
  
discr = b\*\*2 - 4 \* a \* c  
print("D = ", round(discr, 2))  
  
if discr > 0:  
 if a == 0:  
 print("Деление на ноль! Ошибка!")  
 else:  
 x1 = (-b + math.sqrt(discr)) / (2 \* a)  
 x2 = (-b - math.sqrt(discr)) / (2 \* a)  
 print("x1 = ", round(x1, 2),  
 "x2 = ", round(x2, 2))  
elif discr == 0:  
 x = -b / (2 \* a)  
 print("x = ", round(x, 2))  
else:  
 print("Корней нет.")

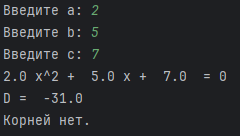
**Задание №2:**

При введении параметров, которые приводят к дискриминанту больше нуля результат следующий:  
 

При попытке деления на ноль:



При отсутствии корней:



**Контрольные вопросы:**

1. Оценка стоимости и причины ошибок в программном обеспечении.
2. Виды и методы тестирования.
3. Понятие теста.
4. Требования к разработке тестовых сценариев.
5. Правила разработки тестовых сценариев.

**Ответы:**

1. Оценка стоимости и причины ошибок в программном обеспечении:

* тестирование программного обеспечения показывает наличие ошибок, но не доказывает их отсутствие;
* цели тестирования программного обеспечения включают выявление дефектов, проверку функциональности, оценку надёжности и производительности, проверку совместимости и подтверждение соответствия стандартам.

1. Виды и методы тестирования:

* модульное тестирование, интеграционное тестирование, системное тестирование, приёмочное тестирование, нагрузочное тестирование и тестирование совместимости;
* роли и обязанности в тестировании программного обеспечения включают тестировщика, тест-лидера, тест-аналитика, тест-инженера и тестового архитектора.

1. Понятие теста: тест — это инструмент для проверки соответствия программного продукта заданным требованиям и выявления дефектов.
2. Требования к разработке тестовых сценариев:

* сценарии должны быть чёткими и понятными;
* сценарии должны охватывать все возможные варианты использования программного продукта;
* сценарии должны быть независимыми друг от друга.

1. Правила разработки тестовых сценариев:

* сценарии должны быть разработаны на основе требований к программному продукту;
* сценарии должны быть протестированы и утверждены командой разработчиков;
* сценарии должны быть регулярно обновляться и корректироваться по мере изменения требований к продукту.