

Лабораторная работа № 1-2

Разработка тестового сценария проекта

Цель: получить навыки разработки тестовых сценариев.

Теоретические вопросы

- Оценка стоимости и причины ошибок в программном обеспечении.
- Виды и методы тестирования.
- Понятие теста.
- Требования к разработке тестовых сценариев.
- Правила разработки тестовых сценариев.

Задание № 1. Написать программу решения квадратного уравнения $ax^2 + bx + c = 0$.

Задание № 2. Найти минимальный набор тестов для программы нахождения вещественных корней квадратного уравнения $ax^2 + bx + c = 0$. Решение представлено в таблице.

Но- мер теста	a	b	c	Ожидаемый результат	Что проверяется
1	2	-5	2	$x_1=2, x_2=0,5$	Случай вещественных корней
2	3	2	5	Сообщение	Случай комплексных корней
3	3	-12	0	$x_1=4, x_2=0$	Нулевой корень
4	0	0	10	Сообщение	Неразрешимое уравнение
5	0	0	0	Сообщение	Неразрешимое уравнение
6	0	5	17	Сообщение	Неквадратное уравнение
7	9	0	0	$x_1=x_2=0$	Нулевые корни

Таким образом, для этой программы предлагается минимальный набор функциональных тестов, исходя из 7 классов выходных данных.

```
ссылка: 0
static void Main(string[] args)
{
    Console.WriteLine("a=");
    int a = Convert.ToInt32(Console.ReadLine());

    Console.WriteLine("b=");
    int b = Convert.ToInt32(Console.ReadLine());

    Console.WriteLine("c=");
    int c = Convert.ToInt32(Console.ReadLine());

    double d = M

    if (d < 0) C
    else
    {
        double x
        double x
        if (x1 =
        else Con
    }
}
```

Заповеди по отладки программного средства, предложенные Г. Майерсом.

Заповедь 1. Считайте тестирование ключевой задачей разработки ПС, поручайте его самым квалифицированным и одаренным программистам, нежелательно тестировать свою собственную программу.

Заповедь 2. Хорош тот тест, для которого высока вероятность обнаружить ошибку, а не тот, который демонстрирует правильную работу программы.

Заповедь 3. Готовьте тесты как для правильных, так и для неправильных данных.

Заповедь 4. Документируйте пропуск тестов через компьютер, детально изучайте результаты каждого теста, избегайте тестов, пропуск которых нельзя повторить. *Заповедь 5.* Каждый модуль подключайте к программе только один раз, никогда не изменяйте программу, чтобы облегчить ее тестирование.

Заповедь 6. Пропускайте заново все тесты, связанные с проверкой работы какой-либо программы ПС или ее взаимодействия с другими программами, если в нее были внесены изменения (например, в результате устранения ошибки).

Задание № 3. Разработайте набор тестовых сценариев (как позитивных, так и негативных) для следующей программы:

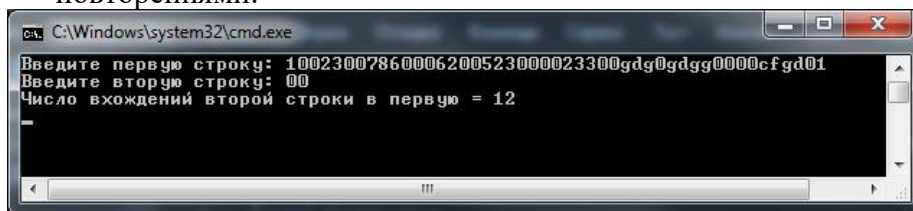
Имеется консольное приложение (разработайте самостоятельно). Ему на вход подается 2 строки. На выходе приложение выдает число вхождений второй строки в первую. Например:

Строка 1	Строка 2	Вывод
абвгабвг	аб	2
стстсап	стс	2

Набор тестовых сценариев запишите в виде таблицы, приведенной выше (в задании 2).

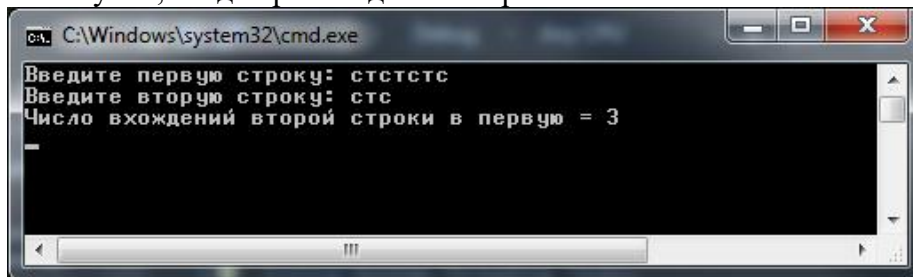
Примеры тестовых сценариев:

1. Случай, когда подстрока входит в первую строку большое количество раз с повторениями:



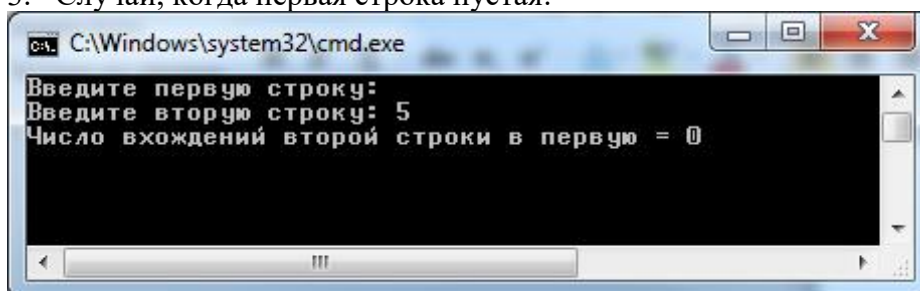
```
C:\Windows\system32\cmd.exe
Введите первую строку: 10023007860006200523000023300gdg0gdgg0000cfd01
Введите вторую строку: 00
Число вхождений второй строки в первую = 12
```

2. Случай, когда происходит повторение с наложением:



```
C:\Windows\system32\cmd.exe
Введите первую строку: стстстс
Введите вторую строку: стс
Число вхождений второй строки в первую = 3
```

3. Случай, когда первая строка пустая:



```
C:\Windows\system32\cmd.exe
Введите первую строку:
Введите вторую строку: 5
Число вхождений второй строки в первую = 0
```

Разобрать минимум 7 тестовых сценариев с разными случаями.

Задание № 4. Оформите отчет с кодами программ, таблицами и скриншотами всех проверяемых тестовых случаев.