

Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации Сибирский Государственный Университет
Телекоммуникаций и Информатики СибГУТИ

Кафедра прикладной математики и кибернетики

Лабораторная работа №3
по дисциплине “Современные технологии программирования”
Модульное тестирование программ на языке C++ в среде Visual Studio

Выполнил:
Студент группы ИП-916
Меньщиков Д.А.

Работу проверил:
Агалаков А. А.

Новосибирск, 2022

Задание:

Разработайте на языке C++ класс, содержащий набор функций в соответствии с вариантом задания.

Разработайте тестовые наборы данных по критерию С2 для тестирования функций класса.

Протестировать функции с помощью средств автоматизации модульного тестирования Visual Studio.

Провести анализ выполненного теста и, если необходимо отладку кода. Функция получает целое числа а. Возвращает число, полученное путём циклического сдвига значения разрядов а на заданное число позиций в заданном направлении.

Функция вычисляет и возвращает число Фибоначчи по его номеру. Функция получает целое числа а, р, n. Возвращает число, полученное путём удаления из а, начиная с позиции р, число разрядов n. Разряды нумеруются слева направо.

Функция получает двумерный массив вещественных переменных А. Отыскивает и возвращает сумму компонентов массива, лежащих выше побочной диагонали с чётной суммой значений индексов.

Выполнение лабораторной работы:

Код программы:

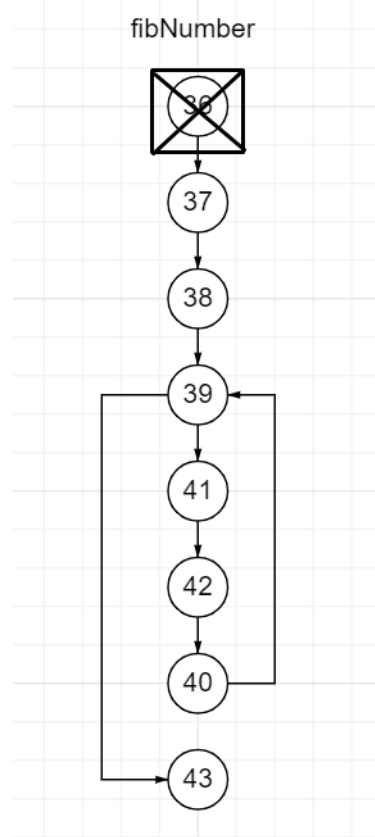
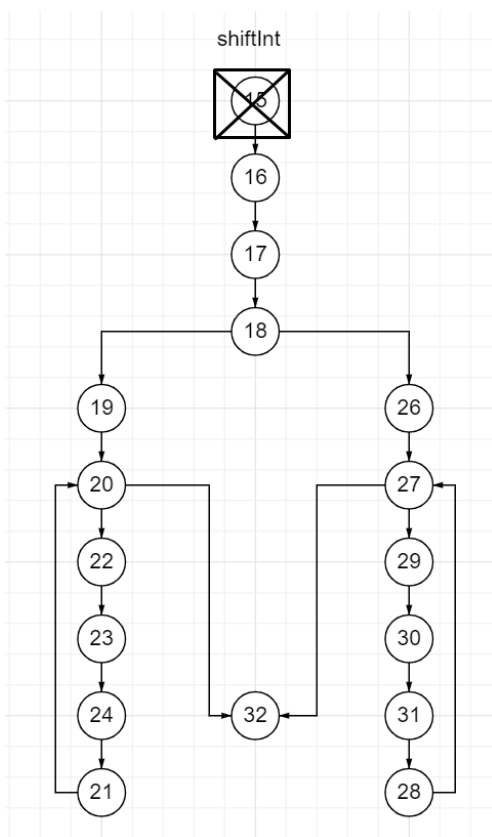
```
15     int shiftInt(int number, int shiftCount) {
16         int rankCount = numberRankCount(number);
17         int position = pow(10, rankCount - 1);
18         if (shiftCount < 0) {
19             for (int i = 0;
20                 i < abs(shiftCount);
21                 i++) {
22                 int tmp = number / position;
23                 number %= position;
24                 number = number * 10 + tmp;
25             }
26         } else {
27             for (int i = 0;
28                 i < abs(shiftCount);
29                 i++) {
30                 int tmp = number % 10;
31                 number /= 10;
32                 number = tmp * position + number;
33             }
34         }
35         return number;
36     }
37
38     int fibNumber(int n) {
39         int a = 0;
40         int b = 1;
41         for (int i = 0;
42             i < n;
43             i++) {
44             a = a + b;
45             b = a - b;
46         }
47         return a;
48     }
49
50     int delDecimalInt(int number, int position, int count) {
51         int rankCount = numberRankCount(number);
52         int zeroCount = pow(10, rankCount - position - count);
53         int lastPart = number % zeroCount;
54         int firstPart = number / pow(10, rankCount - position);
55         return firstPart * zeroCount + lastPart;
56     }
```

```

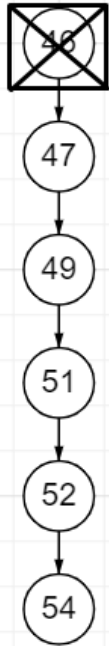
54     int evenSumMatrixEl(std::vector<std::vector<int>> matrix) {
55         int sum = 0;
56         for (int i = 0;
57             i < matrix.size();
58             i++) {
59             if (matrix[i].size() == 0) {
60                 break;}
61             for (int j = 0;
62                 j < matrix[i].size() - i - 1;
63                 j++) {
64                 if (!(i % 2) && !(j % 2)) {
65                     sum += matrix[i][j];}}
66             return sum;
67         }

```

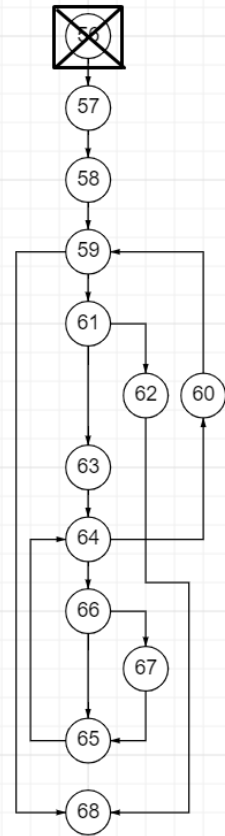
YII:



delDecimallnt



evenSumMatrixEI



Модульные тесты:

```
1 namespace UnitTest1
2 {
3     TEST_CLASS(UnitTest1)
4     {
5     public:
6         Class1 modudeClass;
7         TEST_METHOD(TestShiftIntLeft)
8         {
9             auto actual = modudeClass.shiftInt(12345, -2);
10            auto expected = 34512;
11
12            Assert::AreEqual(expected, actual);
13        }
14
15        TEST_METHOD(TestShiftIntRight)
16        {
17            auto actual = modudeClass.shiftInt(12345, 2);
18            auto expected = 45123;
19
20            Assert::AreEqual(expected, actual);
21        }
22
23        TEST_METHOD(TestShiftIntWithoutShift)
24        {
25            auto actual = modudeClass.shiftInt(12345, 0);
26            auto expected = 12345;
27
28            Assert::AreEqual(expected, actual);
29        }
30
31        TEST_METHOD(TestFibNumber)
32        {
33            auto actual = modudeClass.fibNumber(5);
34            auto expected = 5;
35
36            Assert::AreEqual(expected, actual);
37        }
38
39        TEST_METHOD(TestFibNumberWithZero)
40        {
41            auto actual = modudeClass.fibNumber(0);
42            auto expected = 0;
43
44            Assert::AreEqual(expected, actual);
45        }
46    }
```

```

47     TEST_METHOD(TestDelDecimalInt)
48     {
49         auto actual = modudeClass.delDecimalInt(123456, 2, 2);
50         auto expected = 1256;
51
52         Assert::AreEqual(expected, actual);
53     }
54
55     TEST_METHOD(TestDelDecimalIntWithZero)
56     {
57         auto actual = modudeClass.delDecimalInt(0, 0, 0);
58         auto expected = 0;
59
60         Assert::AreEqual(expected, actual);
61     }
62
63     TEST_METHOD(TestEvenSumMatrixEl)
64     {
65         std::vector<std::vector<int>> matrix{ {1,2,3},{4,5,6},{7,8,9} };
66         auto actual = modudeClass.evenSumMatrixEl(matrix);
67         auto expected = 1;
68
69         Assert::AreEqual(expected, actual);
70     }
71     TEST_METHOD(TestEvenSumMatrixElWithouEl)
72     {
73         std::vector<std::vector<int>> matrix{ { },{ } };
74         auto actual = modudeClass.evenSumMatrixEl(matrix);
75         auto expected = 0;
76
77         Assert::AreEqual(expected, actual);
78     }
79
80     TEST_METHOD(TestNumberRankCount)
81     {
82         auto actual = modudeClass.numberRankCount(123456);
83         auto expected = 6;
84
85         Assert::AreEqual(expected, actual);
86     }
87 };
88 }

```

Обозреватель тестов

10

10

0

Обнаружение тестов завершено: 0. Обнаружено тестов в 893,1 мс

Тестирование	Длительн...	Признаки	Сообщение об ошибке
UnitTest1 (10)	2 мс		
UnitTest1 (10)	2 мс		
UnitTest1 (10)	2 мс		
TestDelDecimalInt	2 мс		
TestDelDecimalIntWithZero	< 1 мс		
TestEvenSumMatrixEl	< 1 мс		
TestEvenSumMatrixElWithoutEl	< 1 мс		
TestFibNumber	< 1 мс		
TestFibNumberWithZero	< 1 мс		
TestNumberRankCount	< 1 мс		
TestShiftIntLeft	< 1 мс		
TestShiftIntRight	< 1 мс		
TestShiftIntWithoutShift	< 1 мс		

Обозреватель тестов

Список ошибок

Вывод