

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
«Владимирский государственный университет
Имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИВлГУ)

Факультет _____ ИТР _____

Кафедра _____ ПИН _____

ЛАБОРАТОРНАЯ РАБОТА №7

по _____ Тестирование ПО _____

Тема _____ Метрики Джилба _____

Руководитель

_____ Колпаков А. А. _____
(фамилия, инициалы)

_____ (подпись) _____ (дата)

Студент ПИН-121 _____
(группа)

_____ Ермилов М.В. _____
(фамилия, инициалы)

_____ (подпись) _____ (дата)

Тема: Метрики Джилба

Задание на лабораторную работу:

					<i>МИВУ.09.03.04</i>			
<i>Из</i>	<i>Лис</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дат</i>				
<i>Разраб.</i>								
<i>Провер.</i>								
<i>Н.контр</i>								
<i>Утв.</i>								
						<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
							2	9
						<i>МИ ВлГУ</i> <i>ПИН - 121</i>		

2.3 Задания для аудиторной работы

Задача 1 «Сложение элементов матриц» (вариант 1)

Создать две матрицы, размер которых вводится с клавиатуры, и заполнить их с помощью датчика случайных чисел. Затем осуществить поэлементное сложение матриц. Исходные и результирующую матрицы вывести на экран монитора, причем отсортировать строки результирующей матрицы по возрастанию. Модифицированную матрицу также вывести на экран монитора. Для разработанной программы на основе лексического анализа исходного текста определить значения метрик Джилба

№ п/п	Операторы, операции	Номера строк	Количество повторений
1	using	1	1
2	class	2	1
3	public static void Main()	4	1
4	int[]	6	1
5	int	7	1
6	char	8	1
7	bool	9	1
8	Random	10	1
9	do	11	1
10	Console.Clear();	13	1
11	Console.WriteLine();	14,15,17,37,41,45,65,69	8
12	...Parse();	16,18	2
13	new int[n][];	19,20,21	3
14	for (...)	22,28,29,31,32,34,35,38,39,46,47,53,54,66,69	15
15	...Next();	30,33	2
16	Console.WriteLine();	40,44,48,68	4
17	while(...)	50,71	2
18	if (...)	56	1
19	=	7,10,16,18,19,20,21,22,24,25,26,28,29,31,32,33,34,35,36,38,39,42,43,46,47,49,52,53,54,58,59,60,66,67,70,71	35
20	+	36,59,60	3
21	<	22,28,29,31,32,34,35,38,39,42,43,49,47,53,54,66,67	17
22	++	22,28,31,32,34,35,38,39,42,43,46,47,53,54,66,67	16
23	Parse	16,70	2

24	;	1,6,8,9,10,13,14,15,16,17,18,19,20,24,25,26,30,33,36,37,40,41,44,45,48,49,52,58,59,60,61,65,68,69,70,71	34
25	,	6,7,38,40,44,46,47,66,67,68	18
26	()	4,10,13, 14,15,16,17,22,28,29,30,31,32,33,34,35,37,38,39, 40, 41,42,43,44,45,46,47,48,53,54,56,65,66,67,68,69,70,71	34
27	[]	6,19,20,21,24,25,26,33,30,36,40,44,48,56,58,59,60,67,68	41

Общее число: 247

					МИВУ.09.03.04	Лист
						3
Из	Лист	№ докум.	Подп.	Дат		

Исходя из полученных данных получаем

следующие результаты оценки характеристик программы:

- число операторов условий LIF равно 1 (таблица 19, п. 18);
- абсолютная сложность $CL = 1$, так как в программе используется один оператор условия;
- относительная сложность программы равна:
 $cl = CL/L = 1/247 = 0,0040$.

С точки зрения лексического анализа исходного текста программы представленное решение не является сложным, так как количество ветвлений в программе весьма невелико (1 оператор условия), что подтверждается невысокой величиной метрики относительной сложности программы.

Задача 2 «Сложение элементов матриц» (вариант 2)

Создать две матрицы, размер которых вводится с клавиатуры, и заполнить их с помощью датчика случайных чисел. Затем осуществить поэлементное сложение матриц. Исходные и результирующую матрицы вывести на экран монитора, причем отсортировать строки результирующей матрицы по возрастанию. Модифицированную матрицу также вывести на экран монитора. Для разработанной программы на основе лексического анализа исходного текста определить значения метрик Джилба.

№п/п	Операторы, операции	Номера строк	Количество повторений
1	using	1	1
2	class	2	1
3	{ }		8
4	public	5	1
5	new	6,9	2
6	for (...)	7,15,16,23,24,29,30,53,54	9
7	...Next()	17,	
8	Console.WriteLine()	22 43 44 46 61	5
9	...Parse()	45,47,62	3
10	Console.Write()	25	1
11	...Sort()	31	1
12	=	15 16 17 23 24 29 30 37 38 45 47 48 49 50 53 54 62	17
13	==	63	2
14	;	1 6 7 11 15 16 17 22 23 24 25 3135 36 37 38 39 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63	49
15	,	4 13 20 23 25 35 36 48 49 50 51 52 56 57 58 60	20
16	()	4 7 13 15 16 17 20 22 23 24 25 29 30 31 33 37 38 42 43 44 45 46 47 48 49 50 51 52 53 54 56 57 58 59 60 61 62 63	34
17	int	4 15 16 23 24 29 30 36 53 54	13

18	Random	38	1
19	...Next()	17	1
20	do... while()	40(63)	1
21	Console.Clear()	42	1

Общее число: 171

Исходя из полученных данных получаем следующие результаты оценки характеристик программы:

- Lloop = 9 - количество циклов;
- число операторов условий L do while равно 1 (таблица 19, п. 18);
- абсолютная сложность CL = 1, так как в программе используется один оператор условия;
- относительная сложность программы равна:
 $cl = CL/L = 1/171 = 0,0058$.
- CL = 9+1=10 - общее количество операторов условий, из них 7 операторов циклов, в которых содержатся условия, и один условный оператор;

Программа с точки зрения метрик Джилба имеет невысокую логическую сложность, так как насыщенность текста операторами условий незначительна.

2.4 Задания для самостоятельной работы

Задача 15. Вычислить сумму цифр натурального числа, задаваемого с клавиатуры.

```
static void Main(string[] args)
{
    Console.WriteLine("Введите натуральное число: ");
    string input = Console.ReadLine();
    int sum = 0;

    foreach (char c in input)
    {
        sum += c - '0';
    }
    Console.WriteLine("Сумма цифр: " + sum);
    Console.ReadKey();
}
```

№ п/п	Операторы, операции	Номера строк	Количество повторений
1	using	1	1
2	namespace	2	1
3	class	4	1
4	{ }	3(20), 5(19), 7(18), 13(15)	4
5	static	6	1

					МИВУ.09.03.04	Лист
						5
Из	Лист	№ докум.	Подп.	Дат		

6	void	6	1
7	(string[] args)	6	1
8	Console.Write	8	1
9	Console.ReadLine	9	1
10	int	10,	4
11	char	12	1
12	foreach	12	1
13	in	12	1
14	'0'	14	1
15	Console.WriteLine	16	1
16	Console.ReadKey	17	1
17	=	9,10	2
18	+=	14	1
19	;	1, 8, 9, 10, 14, 16, 17	7

Метрики Джилба:

CL – абсолютная сложность программы

Абсолютная сложность определяется количеством операторов условий в программе. В данной программе нет операторов условий (if, switch), следовательно, CL = 0.

cl – относительная сложность программы

Относительная сложность вычисляется как отношение абсолютной сложности (CL) к общему числу операторов (L). $cl = CL / L$

Количество операторов цикла (Lloop)

Программа содержит один оператор цикла foreach.

L loop = 1

В программе нет операторов условия.

Число модулей или подсистем (Lmod)

В программе один модуль – основной метод Main.

Lmod = 1

Отношение числа связей между модулями к числу модулей

Программа содержит один модуль, и связи между модулями отсутствуют.

SVmod = 0 / 1 = 0

Отношение числа ненормальных выходов из множества операторов к общему числу операторов.

В данной программе нет ненормальных выходов (исключений).

Надежность программы

Надежность определяется как единица минус отношение числа логических сбоев к общему числу запусков. В данной программе логические сбои маловероятны при корректном вводе данных.

Прецизионность (частота появления ошибок)

Программа корректно обрабатывает корректный ввод, частота ошибок на входе зависит от корректности ввода пользователем.

Прецизионность ≈ 1

Итоговая оценка

					МИВУ.09.03.04	Лист
						6
Из	Лис	№ докум.	Подп.	Дат		

Абсолютная сложность (CL): 0
 Относительная сложность (cl): $0 / 6 \approx 0$
 Количество операторов цикла (Lloop): 1
 Количество операторов условия (LIF): 0
 Число модулей или подсистем (Lmod): 1
 Отношение числа связей между модулями к числу модулей (SVmod): 0
 *Отношение числа ненормальных выходов к общему числу операторов (SV)**: 0
 Надежность программы: 1
 Мера точности (свободы от ошибок): 1
 Прецизионность (частота появления ошибок): 1

Заключение

Программа имеет нулевую абсолютную и относительную сложность, что указывает на ее простоту. Она содержит один цикл и не использует условий, что делает её легко читаемой и понятной. Надежность, мера точности и прецизионность близки к единице, что говорит о высоком качестве и надежности программы при корректном вводе данных.

$$F(x,y)=\begin{cases} x - y, & \text{если } x \geq y; \\ x + y, & \text{если } x < y; \end{cases}$$

Вывести на экран в виде таблицы значения функции $F(x, y)$ для значений аргументов $x = 0,5 - 0,7$ с шагом $0,1$ и $y = 0,2 - 1,0$ с шагом $0,2$.

```

using System;
namespace laba7TEST
{
    internal class Program
    {
        static void Main(string[] args)
        {
            double[] xValues = { 0.5, 0.6, 0.7 };
            double[] yValues = { 0.2, 0.4, 0.6, 0.8, 1.0 };
            Console.WriteLine("x \ y | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 |");
            foreach (double x in xValues)
            {
                Console.Write($" {x}  |");
                foreach (double y in yValues)
                {
                    double result = (x >= y) ? (x - y) : (x + y);
                    Console.Write($" {result:F2} |");
                }
                Console.WriteLine();
            }
            Console.ReadKey();
        }
    }
}
  
```

x \ y	0.2	0.4	0.6	0.8	1.0
0,5	0,30	0,10	1,10	1,30	1,50
0,6	0,40	0,20	0,00	1,40	1,60
0,7	0,50	0,30	0,10	1,50	1,70

Рисунок1 - результат работы программы

№п/п	Операторы, операции	Номера строк	Количество повторений
1	using	1	1
2	namespace	2	1
3	class	3	1
4	{ }	3(24), 5(23), 12(20),15(18)	4
5	static	5	1
6	void	6	1
7	Main	6	1
8	string[]	6	1
9	double[]	8,9	2
10	=	8,9,16	3
11	Console.WriteLine	10,19	2
12	foreach	11,14	2
13	double	8,9, 11, 14,16	3
14	>=	16	1
15	? :	16	1
16	-	16	1
17	+	16	1
18	Console.Write	17,13	2
19	Console.WriteLine	10,19	2
20	Console.ReadKey	21	1
21	;	1, 8,9, 10,13, 16, 17,19,21	9
22	[]	6, 8,9	3
23	()	6,10, 11, 13, 16, 17,19,21	8
24	,	8,9	6
25	"	10,13,17	6
26	\\	10	1

Всего : 64

Оценка по метрикам Джилба

Абсолютная сложность программы (CL):

Количество операторов условий (LIF):

В программе есть один условный оператор: $(x \geq y) ? (x - y) : (x + y);$

L (IF) =1

Общее количество операторов (L):

Считаем все операторы в программе:

Присваивание (=): 5

Циклы (foreach): 2

Условные операторы (?:) : 1

					МИВУ.09.03.04	Лист
						8
Из	Лист	№ докум.	Подп.	Дат		

Вызовы методов (Console.WriteLine, Console.Write, Console.ReadKey): 5

Доступ к массивам ([]): 4

Прочие операторы (using, namespace, class, static, void, Main, double[], string[]): 9

Всего операторов: 26

=26

Общая оценка

Программа по заданным метрикам Джилба имеет следующие характеристики:

Абсолютная сложность (CL): 1

Относительная сложность (cl): 0.038

Количество операторов цикла (Lloop): 2

Количество операторов условия (LIF): 1

Число модулей или подсистем (Lmod): 1

Отношение числа связей между модулями к числу модулей (SVmod): 0

Отношение числа ненормальных выходов к общему числу операторов (SV*): 0

Надежность: 1

Мера точности: 1

Прецизионность: 0 (безошибочное выполнение)

Вывод: получены навыки работы с оценкой программного средства на основе метрик Холстеда.

Вывод

Программа демонстрирует низкую логическую сложность, что делает её легкой для понимания и сопровождения. Высокая надежность и точность показывают, что программа выполнена корректно и устойчиво. Учитывая отсутствие сложных логических конструкций и ненормальных выходов, данная программа представляет собой пример качественного и надёжного кода.

					<i>МИВУ.09.03.04</i>	Лист
						9
Из	Лист	№ докум.	Подп.	Дат		