

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Качество и метрология программного обеспечения»
Тема: Анализ структурной сложности графовых моделей программ

Студент гр. 8303

Ивченко А.А.

Преподаватель

Кирияничков В.А.

Санкт-Петербург

2022

Цель работы.

Изучить структурную сложность графовых моделей программ и метрики для её оценки.

Ход выполнения.

Был выбран 5 вариант задания. Структура управляющего графа для этого варианта представлена на рисунке 1.

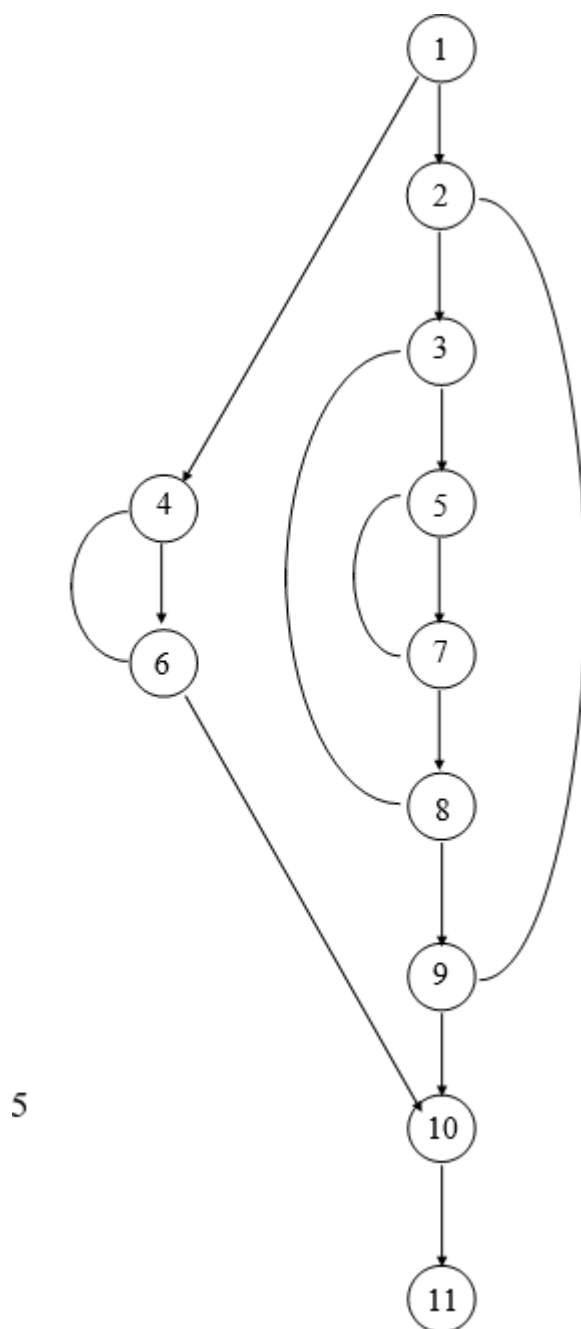


Рисунок 1 – Управляющий граф для 5 варианта

Был выполнен расчёт структурной сложности этой программы.

По первому критерию:

Минимальное количество маршрутов – 2:

- 1 – 4 – 6 – 4 – 6 – 10 – 11
- 1 – 2 – 3 – 5 – 7 – 8 – 9 – 2 – 3 – 5 – 7 – 8 – 3 – 5 – 7 – 5 – 7 – 8 – 9 – 10 – 11

По первому критерию требуется 2 маршрута $M = 2$.

Сложность программы по первому критерию: $S1 = 3 + 10 = 13$.

По второму критерию:

$N = 11$; $Y = 15$; $P = 1$.

$Z = Y - N + 2 * P = 15 - 11 + 2 * 1 = 6$

Линейно-независимые маршруты:

- M1: 1 – 2 – 3 – 5 – 7 – 8 – 9 – 10 – 11
- M2: 1 – 4 – 6 – 10 – 11
- M3: 2 – 3 – 5 – 7 – 8 – 9 – 2
- M4: 3 – 5 – 7 – 8 – 3
- M5: 4 – 6 – 4
- M6: 5 – 7 – 5

$S2 = 1 + 1 + 2 + 3 + 2 + 4 = 13$.

Автоматический расчёт

```
Min ways....
----- Path #1 -----
-> 1 -> 2 -> 3 -> 5 -> 7 -> 5 -> 7 -> 8 -> 3 -> 5 -> 7 -> 8 -> 9 -> 2 -> 3 -> 5
-> 7 -> 8 -> 9 -> 10 -> 11
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 4 -> 6 -> 4 -> 6 -> 10 -> 11
-----Press a key to continue -----

Complexity = 13
Press a key...
```

Рисунок 2 – Расчёт сложности по первому критерию программой ways.exe

```
Z ways....
----- Path #1 -----
-> 4 -> 6 -> 4
-----Press a key to continue -----
----- Path #2 -----
-> 5 -> 7 -> 5
-----Press a key to continue -----
----- Path #3 -----
-> 3 -> 5 -> 7 -> 8 -> 3
-----Press a key to continue -----
----- Path #4 -----
-> 2 -> 3 -> 5 -> 7 -> 8 -> 9 -> 2
-----Press a key to continue -----
----- Path #1 -----
-> 1 -> 2 -> 3 -> 5 -> 7 -> 8 -> 9 -> 10 -> 11
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 4 -> 6 -> 10 -> 11
-----Press a key to continue -----

Complexity = 13
Press a key...
```

Рисунок 3 – Расчёт сложности по второму критерию программой ways.exe

Граф:

Nodes{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

Top{1}

Last{11}

Arcs{

arc(1,2);

arc(1,4);

arc(2,3);

arc(3,5);

arc(4,6);

arc(5,7);

arc(6,4);

arc(6,10);

arc(7,5);

arc(7,8);

arc(8,3);

arc(8,9);

arc(9,2);

arc(9,10);

arc(10,11);

}

Расчеты для графа из лабораторной работы №1

Код программы на языке C представлен в приложении А.

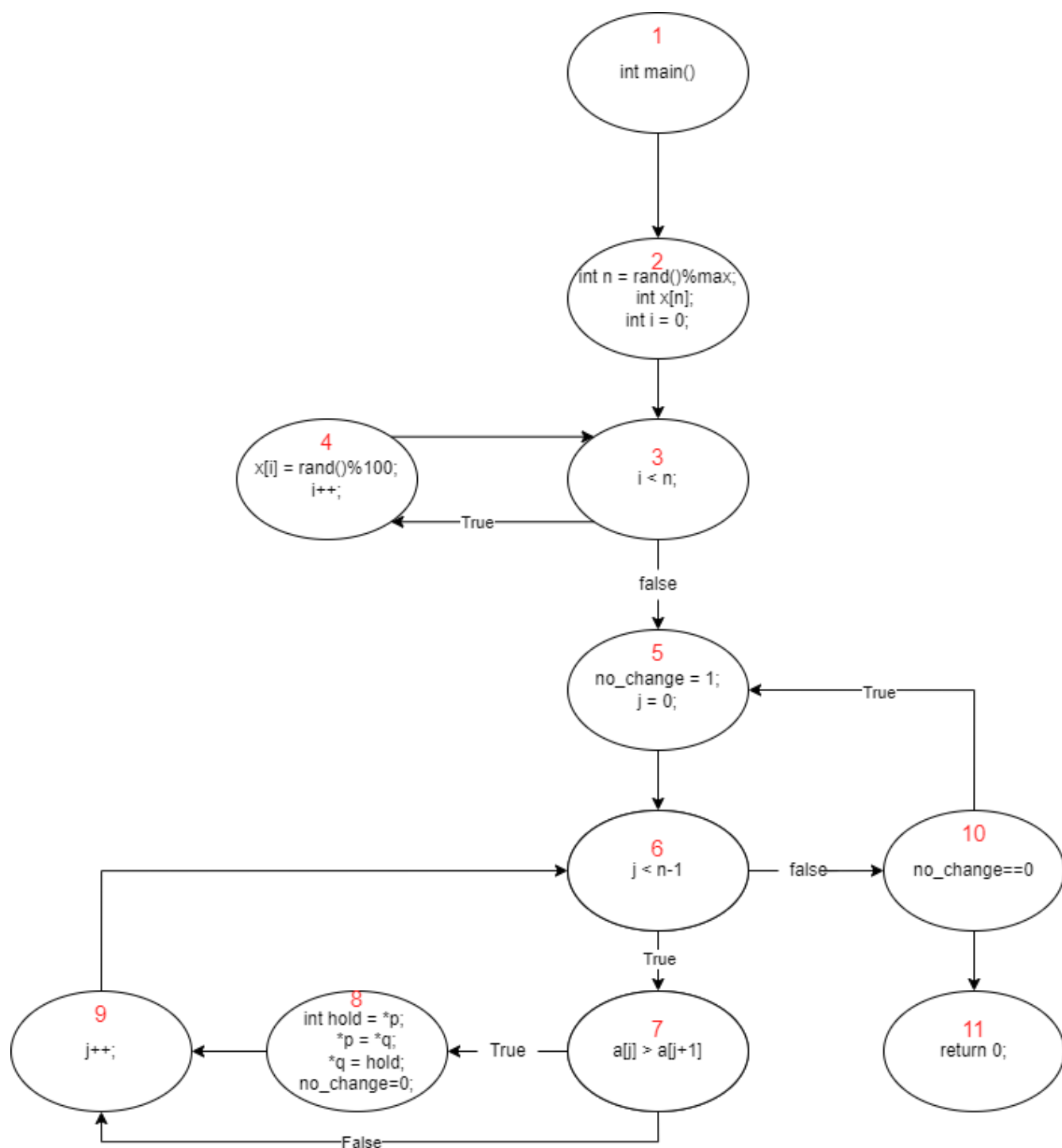


Рисунок 4 – Управляющий граф для программы из лабораторной работы №1

Граф был преобразован. Результат представлен на рисунке 5.

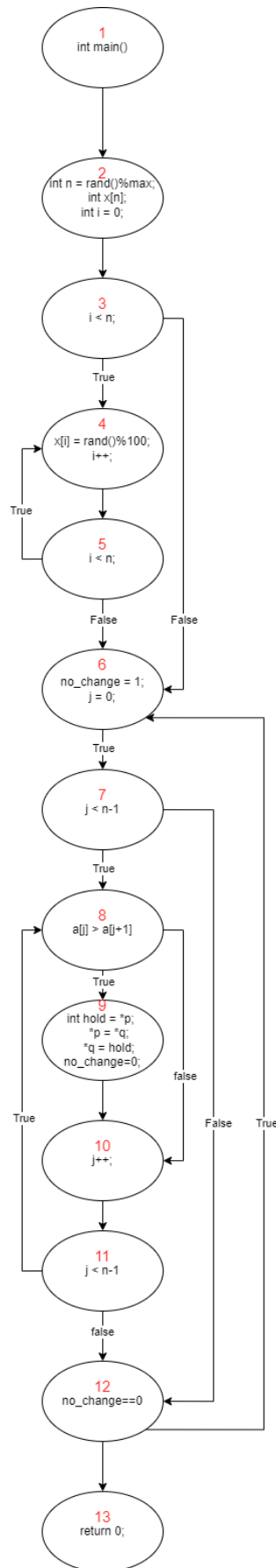


Рисунок 5 – Управляющий граф для программы из лабораторной работы №1 после преобразований

Был выполнен расчёт структурной сложности этой программы по первому критерию – минимальное покрытие вершин и дуг управления.

Минимальное количество маршрутов – 2:

- 1 – 2 – 3 – 6 – 7 – 12 – 13
- 1 – 2 – 3 – 4 – 5 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 8 – 10 – 11 – 12 – 6 – 7 – 12 – 13

В итоге сложность программы $S1 = 3 + 11 = 14$.

Был выполнен расчёт структурной сложности этой программы по второму критерию – каждый линейно-независимый цикл и ациклический участок программы.

$$N = 13; Y = 18; P = 1$$

$$\text{Цикломатическое число } Z = Y - N + 2 \cdot P = 7$$

Необходимые 7 линейно-независимых маршрутов:

- 4 – 5 – 4
- 8 – 10 – 11 – 8
- 6 – 7 – 12 – 6
- 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13
- 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 10 – 11 – 12 – 13
- 1 – 2 – 3 – 4 – 5 – 6 – 7 – 12 – 13
- 1 – 2 – 3 – 6 – 7 – 12 – 13

В итоге сложность программы $S2 = 1 + 3 + 2 + 6 + 6 + 4 + 3 = 25$.

Автоматический расчет

```
lin ways....
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 4 -> 5 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 8 -> 10 ->
11 -> 12 -> 6 -> 7 -> 12 -> 13
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 3 -> 6 -> 7 -> 12 -> 13
-----Press a key to continue -----

Complexity = 14
```

Рисунок 6 – Расчёт сложности по первому критерию программой ways.exe

```
----- Path #1 -----
-> 4 -> 5 -> 4
-----Press a key to continue -----
----- Path #2 -----
-> 8 -> 10 -> 11 -> 8
-----Press a key to continue -----
----- Path #3 -----
-> 6 -> 7 -> 12 -> 6
-----Press a key to continue -----
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 12 -> 13
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 8 -> 10 -> 11 -> 12 -> 13
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 12 -> 13
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 3 -> 6 -> 7 -> 12 -> 13
-----Press a key to continue -----

Complexity = 25
```

Рисунок 7 – Расчёт сложности по второму критерию программой ways.exe

Выводы.

В ходе выполнения лабораторной работы были изучены методы оценки структурной сложности программы на основе управляющего графа, была рассчитана структурная сложность двух программ по двум критериям. Для программы, взятой из первой лабораторной работы, был составлен управляющий граф. Расчёт структурной сложности программ выполнялся двумя методами: вручную и с помощью программы ways.exe. Разница в результатах измерений по первому и второму критерию можно обосновать недостатком первого критерия, тем что не учитывается комбинаторика сочетаний условий на разных участках маршрутов, а также тем, что число линейно-независимых путей больше, чем число путей, необходимых для минимального покрытия вершин и дуг.