

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. 7304

Нгуен К.Х

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

Изучены методы оценки структурной сложности графовых моделей программ.

Постановка задачи.

1. Выполнить оценивание структурной сложности **двух программ** с помощью критериев:
 - Минимального покрытия вершин и дуг графа управления;
 - Выбора маршрутов на основе цикломатического числа графа.
2. Варианты программ:
 - Программа с заданной структурой управляющего графа, выбираемой из файла `zadan_struct.doc` в соответствии с номером в списке группы;
 - Программа из 1-ой лабораторной работы (управляющий граф составить самостоятельно).
3. Оцениваемые характеристики структурной сложности:
 - Число учитываемых маршрутов проверки программы для заданного критерия;
 - Цикломатическое число;
 - Суммарное число ветвлений по всем маршрутам – оценка структурной сложности;

Ход выполнения.

Вариант №10:

- 1. Программа с заданной структурой управляющего графа, структура управляющего графа, выбираемой из файла `zadan_struct.doc`**

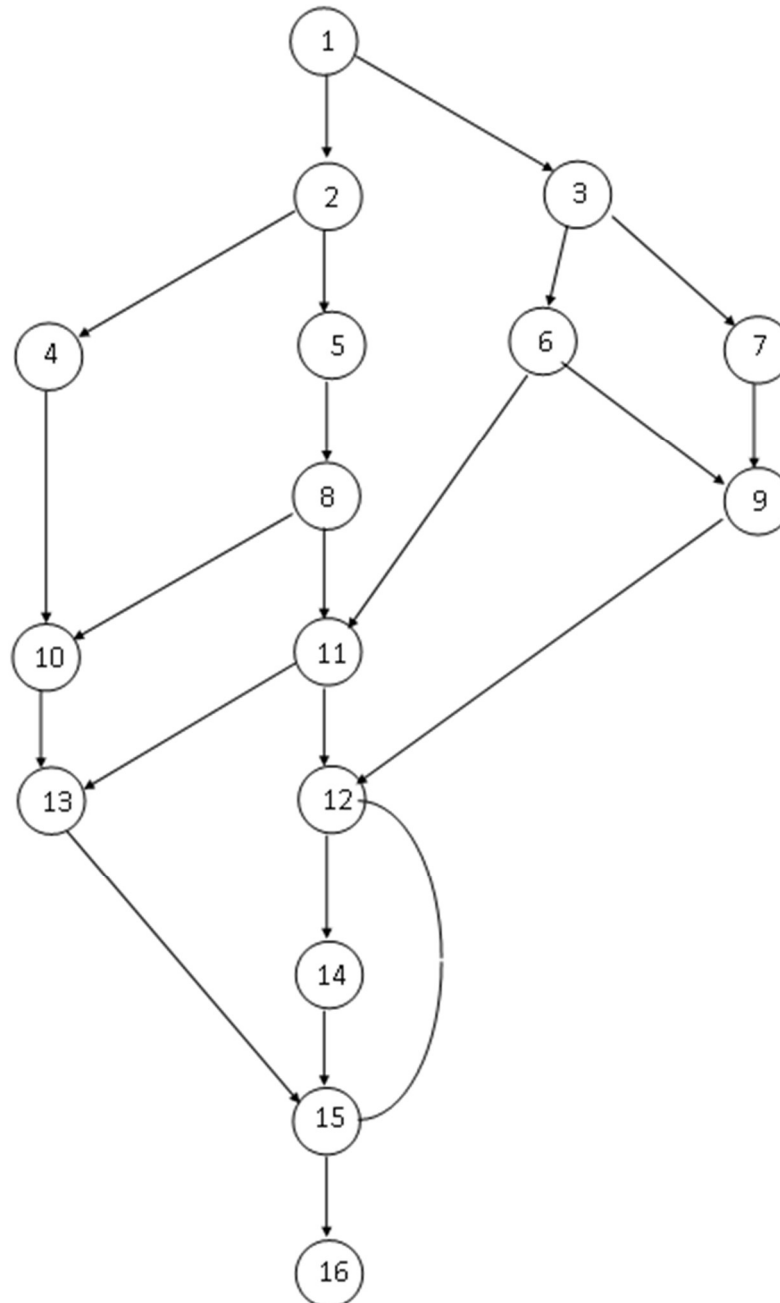


Рисунок 1: Управляющий граф программы для определения структурной сложности

1.1. Анализ структурной сложности графа (без использования программы ways.exe).

1.1.1. Расчет структурной сложности первой программы вручную по первому критерию, а именно по минимальному покрытию вершин и дуг графа управления:

$$M1: \underline{1} - \underline{2} - 5 - \underline{8} - \underline{11} - 12 - 14 - \underline{15} - 12 - 14 - \underline{15} - 16; \quad = 6$$

$$M2: \underline{1} - \underline{2} - 4 - 10 - 13 - \underline{15} - 16; \quad = 3$$

$$M3: \underline{1} - \underline{2} - 5 - \underline{8} - 10 - 13 - \underline{15} - 16; \quad = 4$$

$$M4: \underline{1} - \underline{2} - 5 - \underline{8} - \underline{11} - 13 - \underline{15} - 16; \quad = 5$$

$$M5: \underline{1} - \underline{3} - \underline{6} - \underline{11} - 13 - \underline{15} - 16; \quad = 5$$

$$M6: \underline{1} - \underline{3} - \underline{6} - 9 - 12 - 14 - \underline{15} - 16; \quad = 4$$

$$M7: \underline{1} - \underline{3} - 7 - 9 - 12 - 14 - \underline{15} - 16. \quad = 3$$

Сложность программы: $S = 30$ единиц

1.1.2. Расчет вручную цикломатического числа графа первой программы для дальнейшего применения второго критерия. Для рассматриваемого нами графа $N = 16$ - полное число вершин, $Y = 22$ - количество связывающих дуг и $P = 1$ - число связных компонент

$$Z = Y - N + 2 \times P = 22 - 16 + 2 \times 1 = 8$$

то есть цикломатическое число равно 8.

Построение 8 линейно-независимых циклов и линейно-независимых маршрутов, после чего была подсчитана структурная сложность по второму критерию:

$$M1: 12 - 14 - \underline{15}; \quad = 1$$

$$M2: \underline{1} - \underline{2} - 5 - \underline{8} - \underline{11} - 12 - 14 - \underline{15} - 16; \quad = 5$$

$$M3: \underline{1} - \underline{2} - 4 - 10 - 13 - \underline{15} - 16; \quad = 3$$

$$M4: \underline{1} - \underline{2} - 5 - \underline{8} - 10 - 13 - \underline{15} - 16; \quad = 4$$

$$M5: \underline{1} - \underline{2} - 5 - \underline{8} - \underline{11} - 13 - \underline{15} - 16; \quad = 5$$

$$M6: \underline{1} - \underline{3} - \underline{6} - \underline{11} - 13 - \underline{15} - 16; \quad = 5$$

$$M7: \underline{1} - \underline{3} - \underline{6} - 9 - 12 - 14 - \underline{15} - 16; \quad = 4$$

$$M8: \underline{1} - \underline{3} - 7 - 9 - 12 - 14 - \underline{15} - 16. \quad = 3$$

Сложность программы $S = 30$ единиц.

1.2. Анализ структурной сложности графа (использования программы ways.exe).

Описание графа на входном языке программы анализа имеет вид:

```
Nodes{
  1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16
}

Top{1}
Last{16}

Arcs{
  arc(1,2);
  arc(1,3);
  arc(2,4);
  arc(2,5);
  arc(3,6);
  arc(3,7);
  arc(4,10);
  arc(5,8);
  arc(6,9);
  arc(6,11);
  arc(7,9);
  arc(8,10);
  arc(8,11);
  arc(9,12);
  arc(10,13);
  arc(11,13);
  arc(11,12);
  arc(12,14);
  arc(13,15);
  arc(14,15);
  arc(15,12);
  arc(15,16);
}
```

Результаты анализа графа с помощью программы ways.exe следующие:

```
DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: WAYS

----- Path #1 -----
-> 1 -> 2 -> 4 -> 10 -> 13 -> 15 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 3 -> 6 -> 9 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 5 -> 8 -> 10 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 5 -> 8 -> 11 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #5 -----
-> 1 -> 3 -> 6 -> 11 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #6 -----
-> 1 -> 3 -> 7 -> 9 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #7 -----
-> 1 -> 3 -> 6 -> 11 -> 13 -> 15 -> 16
-----Press a key to continue -----

Complexity = 30
Press a key...
```

Рисунок 2: Программный расчёт структурной сложности первой программы по первому критерию

```
DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: WAYS

Z ways....
----- Path #1 -----
-> 12 -> 14 -> 15 -> 12
-----Press a key to continue -----
----- Path #1 -----
-> 1 -> 2 -> 4 -> 10 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 5 -> 8 -> 10 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 5 -> 8 -> 11 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 5 -> 8 -> 11 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #5 -----
-> 1 -> 3 -> 6 -> 9 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----
----- Path #6 -----
-> 1 -> 3 -> 6 -> 11 -> 13 -> 15 -> 16
-----Press a key to continue -----
----- Path #7 -----
-> 1 -> 3 -> 7 -> 9 -> 12 -> 14 -> 15 -> 16
-----Press a key to continue -----

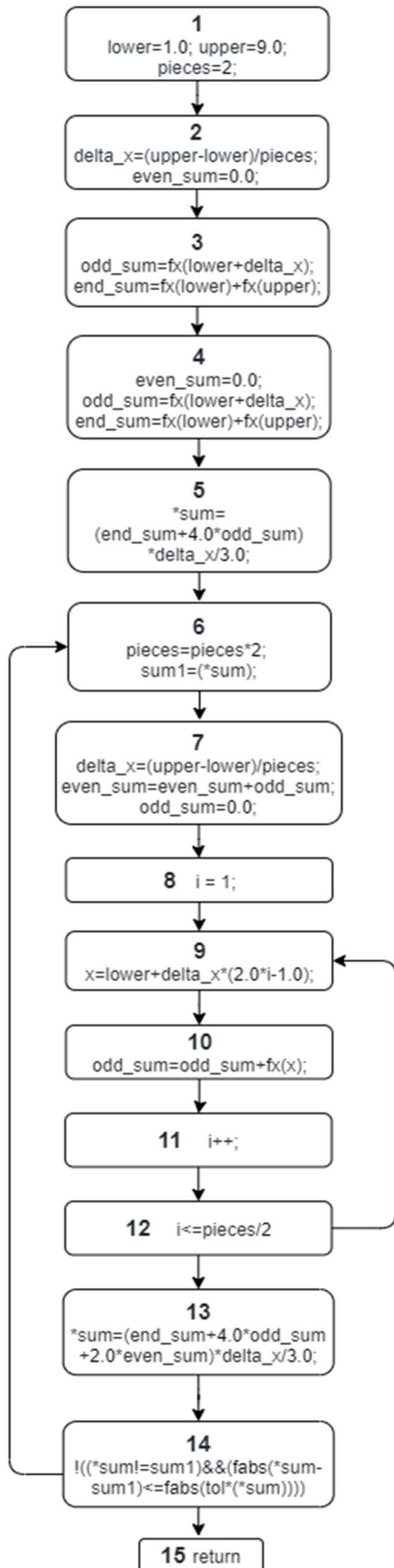
Complexity = 30
Press a key...
```

Рисунок 3: Программный расчёт структурной сложности первой программы по второму критерию

2. Программы из первой лабораторной работы:

2.1. Анализ структурной сложности графа (без использования программы ways.exe).

Был составлен управляющий граф для программы первого лабораторной работы и представлен на Рисунке 4:



Рисунке 4: Управляющий граф программы из первой лабораторной работы для определения структурной сложности

2.1.1. Расчет структурной сложности программы из первой лабораторной работы вручную по первому критерию.

M1: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 9 – 10 – 11 – 12 – 13 – 14
– 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15.

Сложность программы S = 5 единиц.

2.1.2. Расчет вручную цикломатическое число графа программы из первой лабораторной работы для дальнейшего применения второго критерия. Для рассматриваемого нами графа N = 15 - полное число вершин, Y = 16 - количество связывающих дуг и P = 1 - число связных компонент и цикломатическое число

$$Z = Y - N + 2 \times P = 16 - 15 + 2 \times 1 = 3$$

2.1.3. Построение 3 линейно-независимых циклов и линейно-независимых маршрутов, после чего была подсчитана структурная сложность по второму критерию:

M1: 9 – 10 – 11 – 12; = 1

M2: 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14; = 2

M3: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15. = 2

Сложность программы S = 5 единицы

2.2. *Анализ структурной сложности графа (использования программы ways.exe).*

Описание графа на входном языке программы анализа имеет вид:

```
Nodes {  
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15  
}
```

```
Top{1}  
Last{15}
```

```
Arcs{  
arc(1, 2);  
arc(2, 3);
```

```

arc(3,4);
arc(4,5);
arc(5,6);
arc(6,7);
arc(7,8);
arc(8,9);
arc(9,10);
arc(10,11);
arc(11,12);
arc(12,9);
arc(12,13);
arc(13,14);
arc(14,6);
arc(14,15);
}

```

Результаты анализа графа с помощью программы ways.exe следующие:

```

DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: WAYS
Min ways....
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 12 -> 9 -> 10 -> 11
-> 12 -> 13 -> 14 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
Complexity = 5
Press a key...

```

Рисунок 5: Программный расчёт структурной сложности программы из первой лабораторной работы по первому критерию

```

DOSBox 0.74-2, Cpu speed: 3000 cycles, Frameskip 0, Program: WAYS
Z ways....
----- Path #1 -----
-> 9 -> 10 -> 11 -> 12 -> 9
-----Press a key to continue -----
----- Path #2 -----
-> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 12 -> 13 -> 14 -> 6
-----Press a key to continue -----
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 12 -> 13 -> 14 -> 1
5
-----Press a key to continue -----
Complexity = 5
Press a key...

```

Рисунок 6: Программный расчёт структурной сложности программы из первой лабораторной работы по второму критерию

Выводы.

В ходе выполнения лабораторной работы были изучены методы оценки структурной сложности программы на основе его управляющего графа. Структурная сложность двух программ была рассчитана с помощью двух критериев:

- минимального покрытия вершин и дуг графа управления;
- выбора маршрутов на основе цикломатического числа графа.

В результате получены оценки характеристики структурной сложности, совпадающие ручным и программным.