

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

Студентка гр. 8304

Сани З.Б.

Преподаватель

Кирияничков В. А.

Санкт-Петербург

2022

Цель работы.

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

Ход выполнения.

Был выбран 17 вариант задания. Структура управляющего графа для этого варианта представлена на рисунке 1.

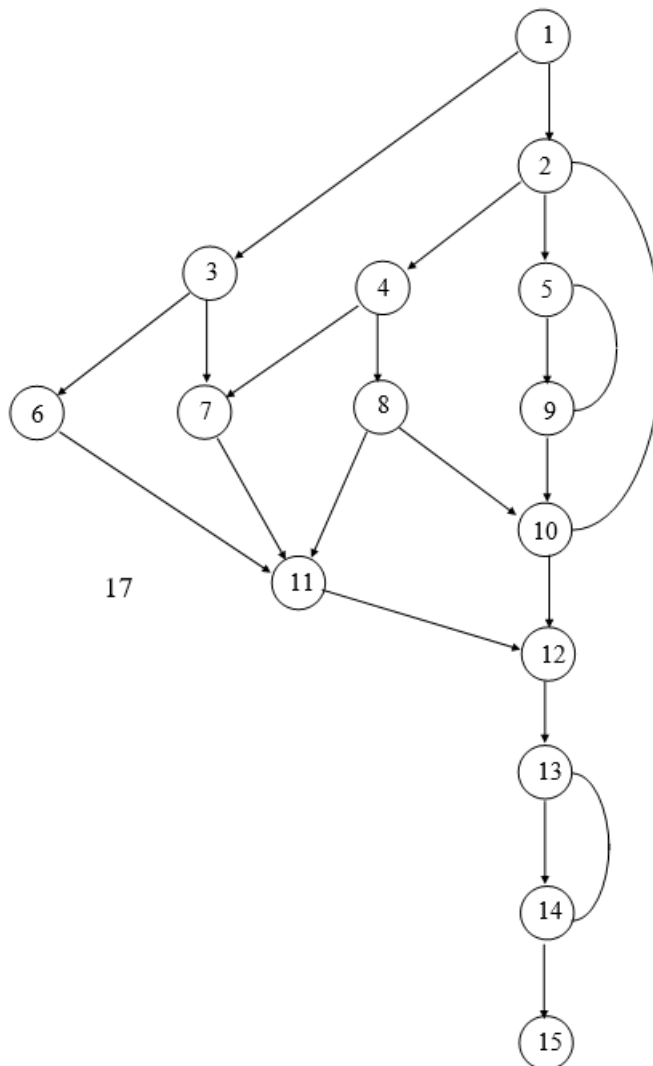
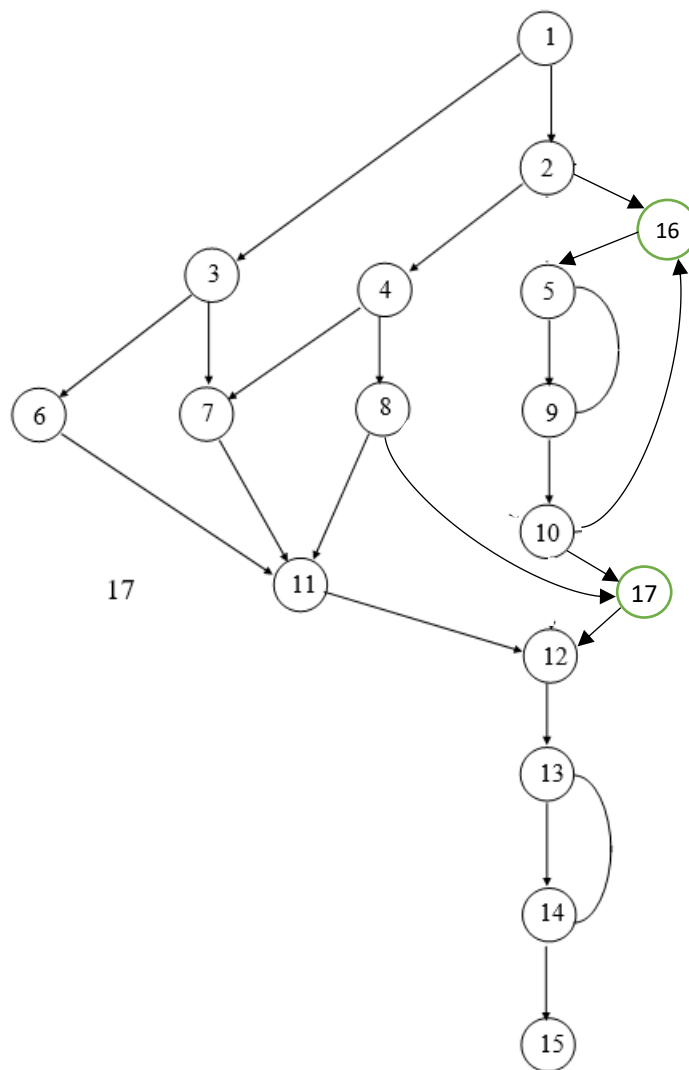


Рисунок 1 – Управляющий граф для 5 варианта

Для использования программы ways.exe граф преобразован к следующему виду:



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

Минимальное количество маршрутов – 6:

- 1–3–6–11–12–13–~~14~~–13–~~14~~–15
- 1–3–7–11–12–13–~~14~~–15
- 1–2–~~4~~–7–11–12–13–~~14~~–15
- 1–2–~~4~~–8–11–12–13–~~14~~–15
- 1–2–~~4~~–8–17–12–13–~~14~~–15
- 1–2–16–5–~~9~~–5–~~9~~–10–16–5–~~9~~–10–17–12–13–~~14~~–15

В итоге сложность программы $S_1 = 4 + 3 + 4 + 5 + 5 + 8 = \underline{29}$.

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

Полное число вершин $N = 17$

Количество связывающих дуг $Y = 24$

Число связанных компонент $P = 1$ (максимально связанный граф получается при добавлении дуги $15 - 1$)

Цикломатическое число $Z = Y - N + 2 \cdot P = 9$

Необходимые 8 линейно-независимых маршрутов:

- 5–9
- 16–5–9–10
- 13–14
- 1–3–6–11–12–13–14–15
- 1–3–7–11–12–13–14–15
- 1–2–4–7–11–12–13–14–15
- 1–2–4–8–11–12–13–14–15
- 1–2–4–8–17–12–13–14–15
- 1–2–16–5–9–5–10–17–12–13–14–15

В итоге сложность программы $S_2 = 1 + 2 + 1 + 3 + 3 + 4 + 5 + 5 + 5 = \underline{29}$.

Для выполнения автоматического расчёта с помощью программы `ways.exe` был подготовлен специальный файл *nodes1.txt*, задающий граф программы. Этот файл представлен в Приложении А. Результат работы программы `ways.exe` представлен на рисунках 2, 3.

```

DOSBox 0.74-3, Cpu speed: max 100% cycles, Frameskip 0, Prog...
Min ways....
----- Path #1 -----
-> 1 -> 2 -> 4 -> 7 -> 11 -> 12 -> 13 -> 14 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 3 -> 6 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 16 -> 5 -> 9 -> 5 -> 9 -> 10 -> 16 -> 5 -> 9 -> 10 -> 17 -> 12 ->
13 -> 14 -> 15
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 4 -> 8 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #5 -----
-> 1 -> 2 -> 4 -> 8 -> 17 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #6 -----
-> 1 -> 3 -> 7 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----

Complexity = 29
Press a key...

```

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

```

DOSBox 0.74-3, Cpu speed: max 100% cycles, Frameskip 0, Prog...
2 ways....
----- Path #1 -----
-> 5 -> 9 -> 5
-----Press a key to continue -----
----- Path #2 -----
-> 16 -> 5 -> 9 -> 10 -> 16
-----Press a key to continue -----
----- Path #3 -----
-> 13 -> 14 -> 13
-----Press a key to continue -----
----- Path #1 -----
-> 1 -> 2 -> 4 -> 7 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 4 -> 8 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 4 -> 8 -> 17 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 16 -> 5 -> 9 -> 10 -> 17 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----

----- Path #3 -----
-> 1 -> 2 -> 4 -> 8 -> 17 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 16 -> 5 -> 9 -> 10 -> 17 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #5 -----
-> 1 -> 3 -> 6 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----
----- Path #6 -----
-> 1 -> 3 -> 7 -> 11 -> 12 -> 13 -> 14 -> 15
-----Press a key to continue -----

Complexity = 29
Press a key...

```

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

Был получен граф потока управления, представленный на рисунке 4, с указанием кода, из которого были получены вершины графа.

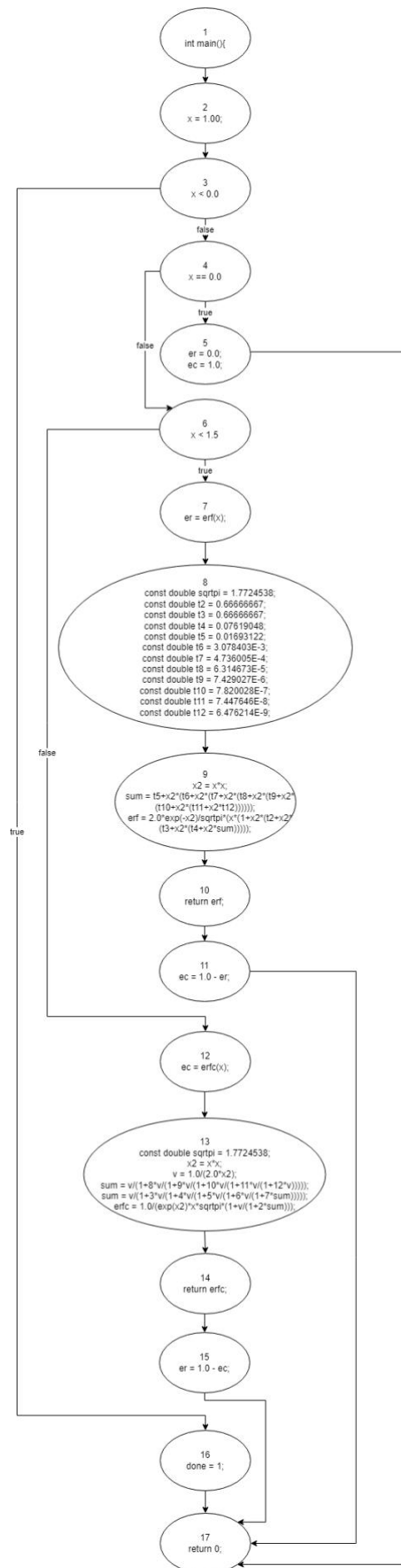


Рисунок 4 – Управляющий граф для программы из лабораторной работы №1

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

Минимальное количество маршрутов – 4:

- 1–2–~~3~~–~~4~~–5–17
- 1–2–~~3~~–~~4~~–~~6~~–7–8–9–10–11–17
- 1–2–~~3~~–~~4~~–~~6~~–12–13–14–15–17
- 1–2–~~3~~–16–17

В итоге сложность программы $S_1 = 2 + 3 + 3 + 1 = \underline{9}$.

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

Полное число вершин $N = 17$

Количество связывающих дуг $Y = 19$

Число связанных компонент $P = 1$ (максимально связанный граф получается при добавлении дуги $17 - 1$)

Цикломатическое число $Z = Y - N + 2 \cdot P = 4$

Необходимые 4 линейно-независимых маршрута:

- 1–2–~~3~~–~~4~~–5–17
- 1–2–~~3~~–~~4~~–~~6~~–7–8–9–10–11–17
- 1–2–~~3~~–~~4~~–~~6~~–12–13–14–15–17
- 1–2–~~3~~–16–17

В итоге сложность программы $S_2 = 2 + 3 + 3 + 1 = \underline{9}$.

Для выполнения автоматического расчёта с помощью программы `ways.exe` был подготовлен специальный файл `nodes2.txt`, задающий граф программы. Этот файл представлен в Приложении Б. Результат работы программы `ways.exe` представлен на рисунках 5, 6.

```
DOSBox 0.74-3, Cpu speed: max 100% cycles, Frameskip 0, Prog...
Min ways....
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 17
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 3 -> 16 -> 17
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 3 -> 4 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 17
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 3 -> 4 -> 6 -> 12 -> 13 -> 14 -> 15 -> 17
-----Press a key to continue -----

Complexity = 9
Press a key...
-
```

Рисунок 5 – Расчёт сложности по первому критерию программой ways.exe

```
DOSBox 0.74-3, Cpu speed: max 100% cycles, Frameskip 0, Prog...
Z ways....
----- Path #1 -----
-> 1 -> 2 -> 3 -> 4 -> 5 -> 17
-----Press a key to continue -----
----- Path #2 -----
-> 1 -> 2 -> 3 -> 4 -> 6 -> 7 -> 8 -> 9 -> 10 -> 11 -> 17
-----Press a key to continue -----
----- Path #3 -----
-> 1 -> 2 -> 3 -> 4 -> 6 -> 12 -> 13 -> 14 -> 15 -> 17
-----Press a key to continue -----
----- Path #4 -----
-> 1 -> 2 -> 3 -> 16 -> 17
-----Press a key to continue -----

Complexity = 9
Press a key...
```

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

Выводы.

В ходе выполнения лабораторной работы была изучена структурная сложность графовых моделей программ и метрики для её оценки. 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.

ПРИЛОЖЕНИЯ

Приложение А. Описание первого графа на входном языке программы анализа.

```
Nodes{
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
}

Top{1}
Last{15}

Arcs{
arc(1,2);
arc(1,3);
arc(2,4);
arc(2,16);
arc(16,5);
arc(3,6);
arc(3,7);
arc(4,7);
arc(4,8);
arc(5,9);
arc(6,11);
arc(7,11);
arc(8,11);
arc(8,17);
arc(9,10);
arc(9,5);
arc(10,16);
arc(10,17);
arc(17,12);
arc(11,12);
arc(12,13);
arc(13,14);
arc(14,13);
arc(14,15);
}
```

Приложение Б. Описание второго графа на входном языке программы анализа.

```
Nodes{  
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17  
}  
  
Top{1}  
Last{17}  
  
Arcs{  
arc(1,2);  
arc(2,3);  
arc(3,4);  
arc(3,16);  
arc(4,5);  
arc(4,6);  
arc(5,17);  
arc(6,7);  
arc(6,12);  
arc(7,8);  
arc(8,9);  
arc(9,10);  
arc(10,11);  
arc(11,17);  
arc(12,13);  
arc(13,14);  
arc(14,15);  
arc(15,17);  
arc(16,17);  
}
```