**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 |  | Гришин К. И. |
| Преподаватель |  | Кирьянчиков В. А. |

Санкт-Петербург

2022

# Цель работы

Изучить метрики оценки графовых моделей программ, а также оценить структурную сложность программ.

# Задание

Вариант 3 (рис.1).

13

12

11

10

9

6

5

3

2

1

4

7

8

14

15

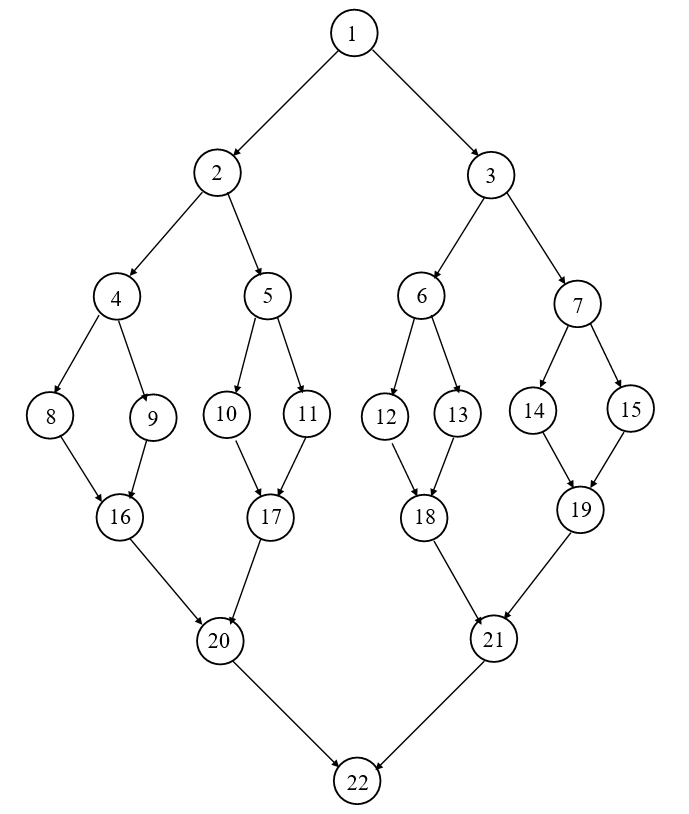


Рисунок . Управляющий граф для 3 варианта.

13

12

11

10

9

6

5

3

2

1

4

7

8

14

15

13

12

11

10

9

6

5

3

2

1

4

7

8

14

15

# Ход выполнения

## Заданный вариантом граф

Для расчета структурной сложности программы по первому критерию необходимо подсчитать минимальное число возможных маршрутов.

1. **1** – **2** – **4** – 8 – 16 – 20 – 22
2. **1** – **2** – **4** – 9 – 16 – 20 – 22
3. **1** – **2** – **5** – 10 – 17 – 20 – 22
4. **1** – **2** – **5** – 11 – 17 – 20 – 22
5. **1** – **3** – **6** – 12 – 18 – 21 – 22
6. **1** – **3** – **6** – 13 – 18 – 21 – 22
7. **1** – **3** – **7** – 14 – 19 – 21 – 22
8. **1** – **3** – **7** – 15 – 19 – 21 – 22

Для подсчета структурной сложности по второму критерию найдены линейно-независимые циклические маршруты. Сначала определено цикломатическое число:

* Полное число вершин – 22
* Число связывающих дуг – 28
* Число связных компонент – 1

Линейно-независимые маршруты:

1. **1** – **2** – **4** – 8 – 16 – 20 – 22
2. **1** – **2** – **4** – 9 – 16 – 20 – 22
3. **1** – **2** – **5** – 10 – 17 – 20 – 22
4. **1** – **2** – **5** – 11 – 17 – 20 – 22
5. **1** – **3** – **6** – 12 – 18 – 21 – 22
6. **1** – **3** – **6** – 13 – 18 – 21 – 22
7. **1** – **3** – **7** – 14 – 19 – 21 – 22
8. **1** – **3** – **7** – 15 – 19 – 21 – 22

Проведен автоматический подсчет сложности программы по обоим критериям с помощью программного средства. Для этого граф представлен в текстовом виде (прил. А).

Результат работы программы:

1. Min ways....
2. -------------- Path #1 ---------------
3. -> 1 -> 2 -> 4 -> 8 -> 16 -> 20 -> 22
4. ---------Press a key to continue ---------
5. -------------- Path #2 ---------------
6. -> 1 -> 3 -> 7 -> 15 -> 19 -> 21 -> 22
7. ---------Press a key to continue ---------
8. -------------- Path #3 ---------------
9. -> 1 -> 2 -> 5 -> 10 -> 17 -> 20 -> 22
10. ---------Press a key to continue ---------
11. -------------- Path #4 ---------------
12. -> 1 -> 2 -> 4 -> 9 -> 16 -> 20 -> 22
13. ---------Press a key to continue ---------
14. -------------- Path #5 ---------------
15. -> 1 -> 2 -> 5 -> 11 -> 17 -> 20 -> 22
16. ---------Press a key to continue ---------
17. -------------- Path #6 ---------------
18. -> 1 -> 3 -> 6 -> 13 -> 18 -> 21 -> 22
19. ---------Press a key to continue ---------
20. -------------- Path #7 ---------------
21. -> 1 -> 3 -> 7 -> 14 -> 19 -> 21 -> 22
22. ---------Press a key to continue ---------
23. -------------- Path #8 ---------------
24. -> 1 -> 3 -> 6 -> 12 -> 18 -> 21 -> 22
25. ---------Press a key to continue ---------
27. Complexity = 24
28. Press a key...
30. Z ways....
31. -------------- Path #1 ---------------
32. -> 1 -> 2 -> 4 -> 8 -> 16 -> 20 -> 22
33. ---------Press a key to continue ---------
34. -------------- Path #2 ---------------
35. -> 1 -> 2 -> 4 -> 9 -> 16 -> 20 -> 22
36. ---------Press a key to continue ---------
37. -------------- Path #3 ---------------
38. -> 1 -> 2 -> 5 -> 10 -> 17 -> 20 -> 22
39. ---------Press a key to continue ---------
40. -------------- Path #4 ---------------
41. -> 1 -> 2 -> 5 -> 11 -> 17 -> 20 -> 22
42. ---------Press a key to continue ---------
43. -------------- Path #5 ---------------
44. -> 1 -> 3 -> 7 -> 15 -> 19 -> 21 -> 22
45. ---------Press a key to continue ---------
46. -------------- Path #6 ---------------
47. -> 1 -> 3 -> 7 -> 14 -> 19 -> 21 -> 22
48. ---------Press a key to continue ---------
49. -------------- Path #7 ---------------
50. -> 1 -> 3 -> 6 -> 13 -> 18 -> 21 -> 22
51. ---------Press a key to continue ---------
52. -------------- Path #8 ---------------
53. -> 1 -> 3 -> 6 -> 12 -> 18 -> 21 -> 22
54. ---------Press a key to continue ---------
56. Complexity = 24
57. Press a key...

Результаты совпадают, что говорит о верности ручных расчетов.

## Структурная сложность программы из первой лабораторной

Для программы из лабораторной получен граф потока (рис. 2)

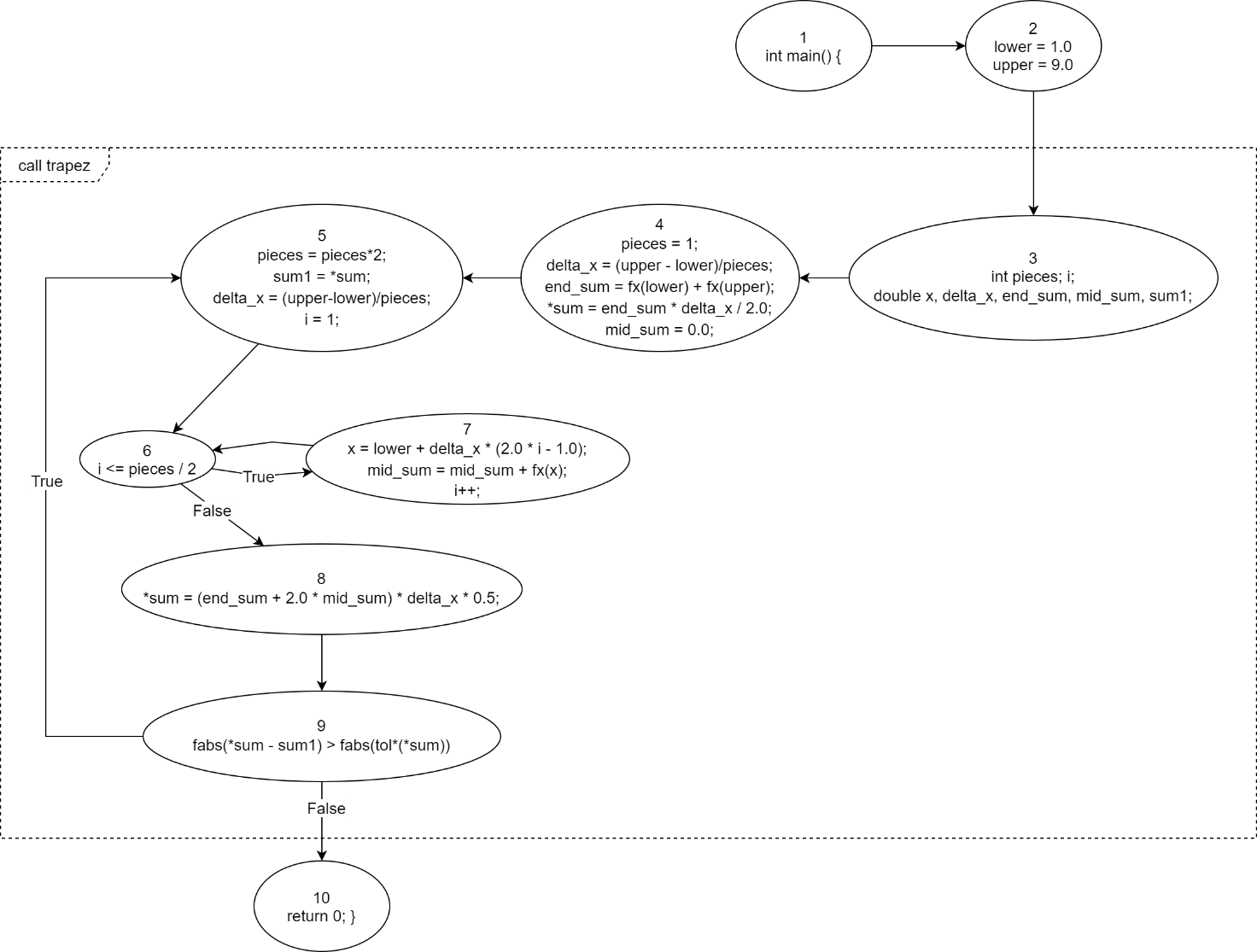


Рисунок . Граф потока программы из лабораторной 1.

Данный граф не подходит программного средства, поэтому его необходимо преобразовать (рис. 3).

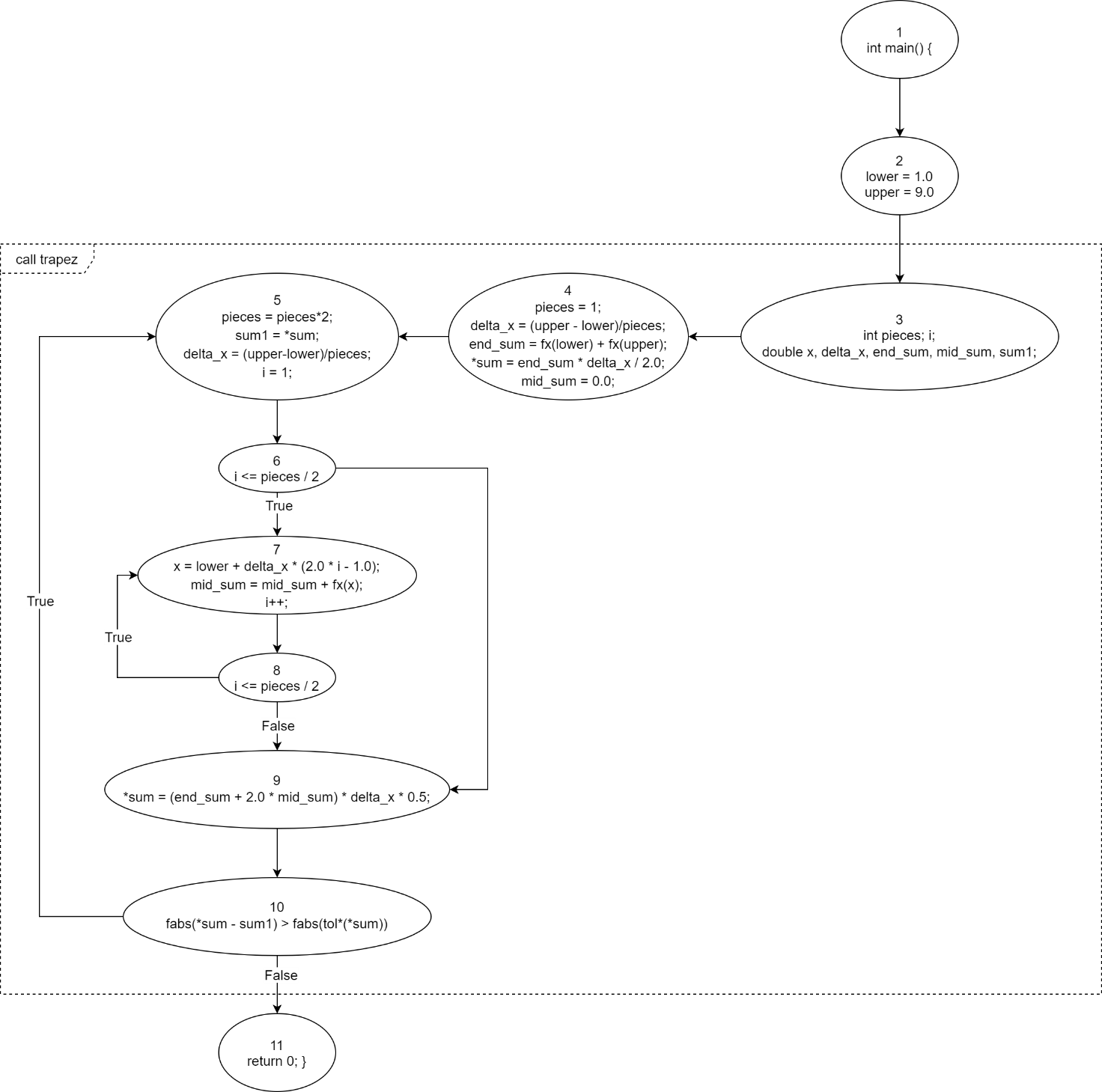


Рисунок . Преобразованный граф.

Для расчета структурной сложности программы по первому критерию необходимо подсчитать минимальное число возможных маршрутов.

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

Для подсчета структурной сложности по второму критерию найдены линейно-независимые циклические маршруты. Сначала определено цикломатическое число:

* Полное число вершин – 11
* Число связывающих дуг – 13
* Число связных компонент – 1

Линейно-независимые циклические маршруты:

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

Проведен автоматический подсчет сложности программы по обоим критериям с помощью программного средства. Для этого граф представлен в текстовом виде (прил. Б).

Результат работы программы:

1. Min ways....
2. -------------- Path #1 ---------------
3. – 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 7 – 8 – 9 – 10 – 5 – 6 – 9 – 10 – 11
4. ---------Press a key to continue ---------
6. Complexity = 6
7. Press a key...
9. Z ways....
10. -------------- Path #1 ---------------
11. – 7 – 8 – 7
12. ---------Press a key to continue ---------
13. -------------- Path #2 ---------------
14. – 5 – 6 – 9 – 10 – 5
15. ---------Press a key to continue ---------
16. -------------- Path #1 ---------------
17. – 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11
18. ---------Press a key to continue ---------
19. -------------- Path #2 ---------------
20. – 1 – 2 – 3 – 4 – 5 – 6 – 9 – 10 – 11
21. ---------Press a key to continue ---------
23. Complexity = 8

Результаты совпадают, что говорит о верности ручных расчетов.

# Вывод

В ходе выполнения лабораторной работы были изучены методы оценки структурной сложности программы на основе управляющего графа.

Рассчитана структурная сложность двух программ по двум критериям:

* 1 критерий: на основе минимального количества маршрутов, проходящих через все дуги и вершины.
* 2 критерий: на основе базовых циклических и ациклических маршрутов.

При отсутствии циклов, структурная сложность не изменяется от выбора критерия, поскольку покрывающие маршруты и ЛНЗ циклические маршруты одинаковы.

Для программы, взятой из первой лабораторной работы, был составлен управляющий граф.

Расчет структурной сложности выполнялся как вручную, так и с помощью предоставленного программного средства «*ways.exe*».

# Приложение А

1. Nodes{
2. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22
3. }
5. Top{1}
6. Last{22}
8. Arcs{
9. arc(1,2);
10. arc(2,4);
11. arc(4,8);
12. arc(8,16);
13. arc(16,20);
14. arc(20,22);
16. arc(4,9);
17. arc(9,16);
19. arc(2,5);
20. arc(5,10);
21. arc(10,17);
22. arc(17,20);
24. arc(5,11);
25. arc(11,17);
27. arc(1,3);
28. arc(3,7);
29. arc(7,15);
30. arc(15,19);
31. arc(19,21);
32. arc(21,22);
34. arc(7,14);
35. arc(14,19);
37. arc(3,6);
38. arc(6,13);
39. arc(13,18);
40. arc(18,21);
42. arc(6,12);
43. arc(12,18);
44. }

# Приложение Б

1. Nodes{
2. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
3. }
4. Top{1}
5. Last{11}
6. Arcs{
7. arc(1,2);
8. arc(2,3);
9. arc(3,4);
10. arc(4,5);
11. arc(5,6);
12. arc(6,7);
13. arc(6,9);
14. arc(7,8);
15. arc(8,9);
16. arc(8,7);
17. arc(9,10);
18. arc(10,11);
19. arc(10,5);
20. }