



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ

по домашнему заданию

по курсу «Анализ Алгоритмов»

на тему: «Графовые модели программ»

Студент группы ИУ7-51Б

(Подпись, дата)

Постнов С. А.

(Фамилия И.О.)

Преподаватель

(Подпись, дата)

Волкова Л. Л.

(Фамилия И.О.)

Преподаватель

(Подпись, дата)

Строганов Ю. В..

(Фамилия И.О.)

Москва — 2023 г.

Содержание

1	Выполнение задания	3
1.1	Реализуемый алгоритм	3
1.2	Выбор языка программирования	3
1.3	Код программы	3
1.4	Модели программ	4
1.4.1	Граф управления программы	5
1.4.2	Информационный граф программы	6
1.4.3	Операционная история программы	7
1.4.4	Информационная история программы	8

1 Выполнение задания

1.1 Реализуемый алгоритм

Расчет значения полинома по схеме Горнера.

1.2 Выбор языка программирования

Для выполнения домашнего задания был выбран язык C++.

1.3 Код программы

Листинг 1.1 – Реализация алгоритма расчета значения полинома по схеме Горнера

```
1 int main() {  
2     srand(time(0));  
3  
4     int power = 2; // (1)  
5     arrayT<double> koefs(power + 1); // (2)  
6  
7     for (int i = 0; i <= power; ++i) { // (3)  
8         koefs[i] = rand() % 10 - 5; // (4)  
9         cout << "Коэффициент при x^"  
10            << power - i << ": "  
11            << koefs[i] << endl;  
12     }  
13  
14     int n = 5; // (5)  
15     arrayT<double> xValues(n); // (6)  
16  
17     for (int i = 0; i < n; ++i) // (7)  
18         xValues[i] = rand() % 10 - 5; // (8)  
19  
20     for (int i = 0; i < n; ++i) { // (9)  
21
```

```

22     double x = xValues[i],           // (10)
23         cur ,                       // (11)
24         prev;                       // (12)
25
26     for (int j = 0; j <= power; ++j) { // (13)
27
28         cur = koefs[j];              // (14)
29
30         if (j)                       // (15)
31             prev = cur + x * prev;   // (16)
32         else
33             prev = cur;              // (17)
34     }
35
36     cout << "Результат для x = "     // (18)
37         << xValues[i] << ": "
38         << prev << endl;
39 }
40 }

```

1.4 Модели программ

На рисунках 1.1–1.4 представлены графовые модели программы.

1.4.1 Граф управления программы

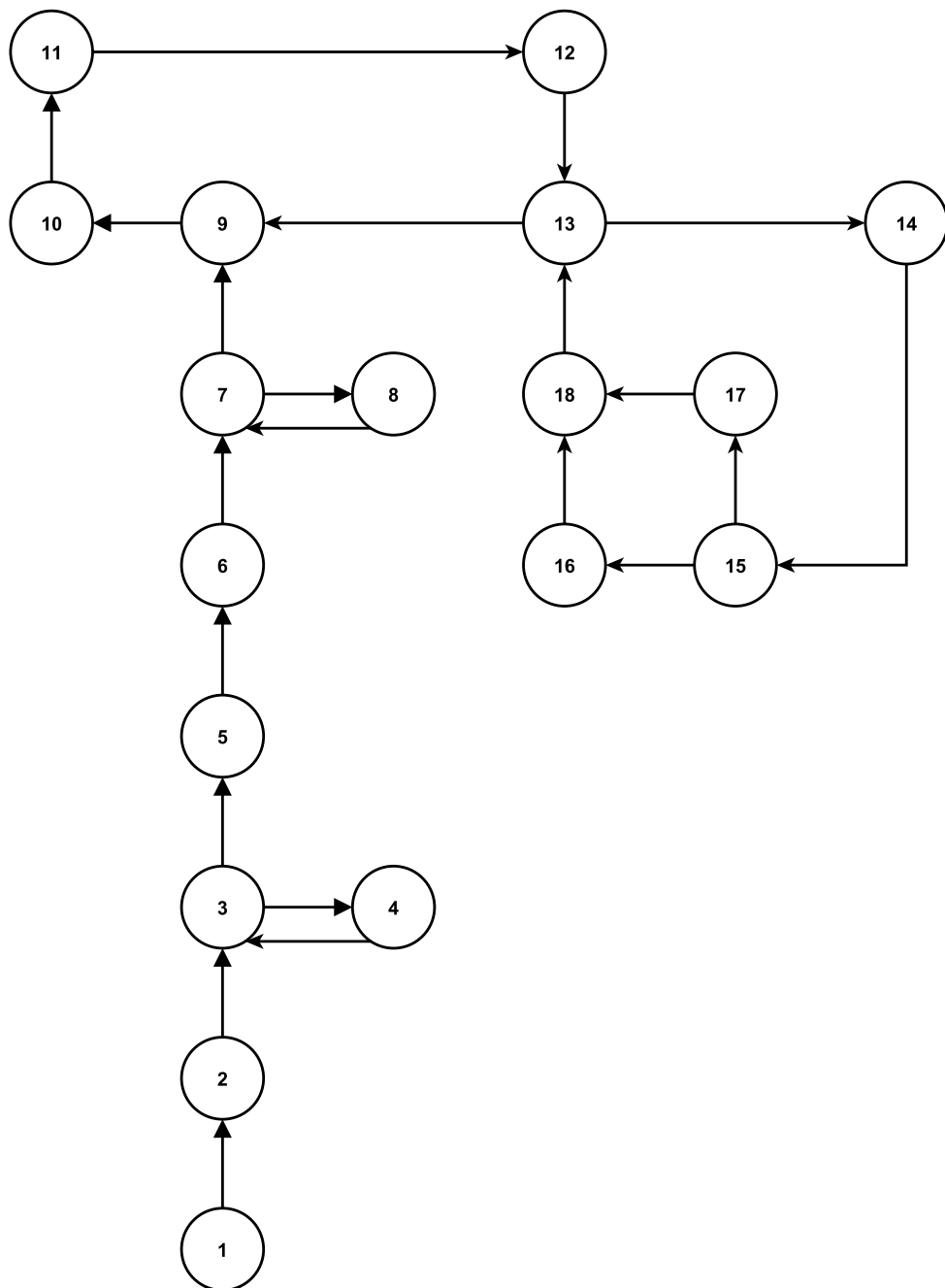


Рисунок 1.1 – Граф управления

1.4.2 Информационный граф программы

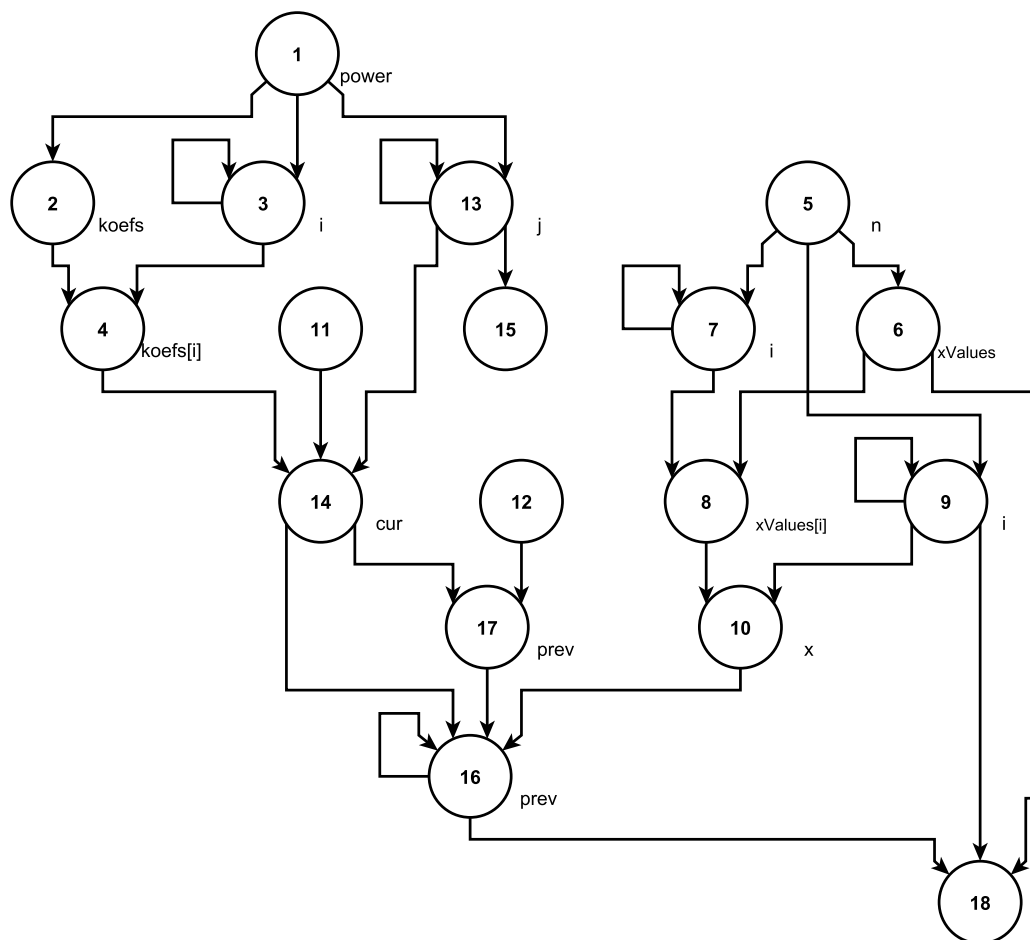


Рисунок 1.2 – Информационный граф

1.4.3 Операционная история программы

Рассчитаем значение полинома вида $x + 1$ для значений $x = 1, 2$.

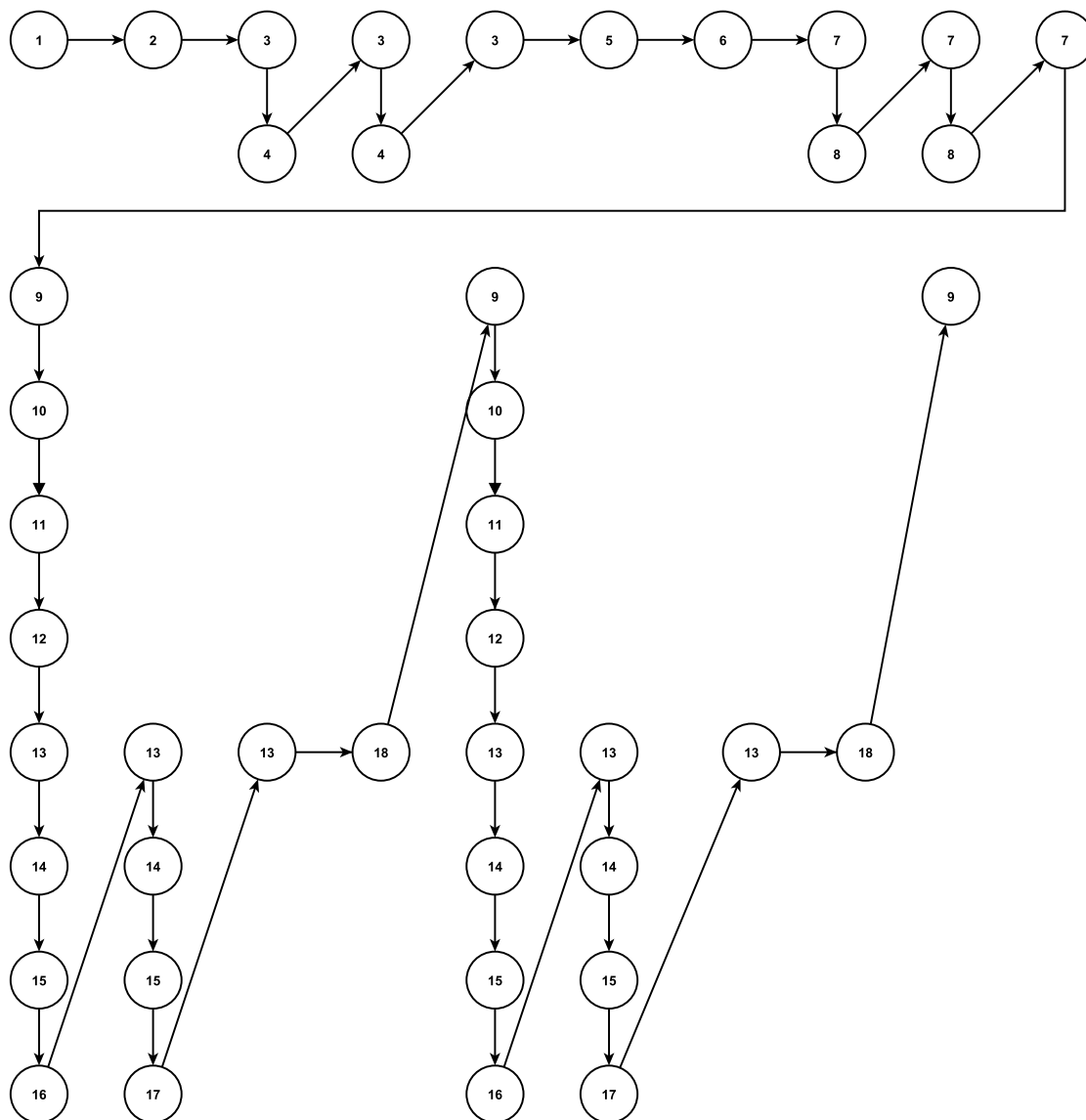


Рисунок 1.3 – Операционная история

1.4.4 Информационная история программы

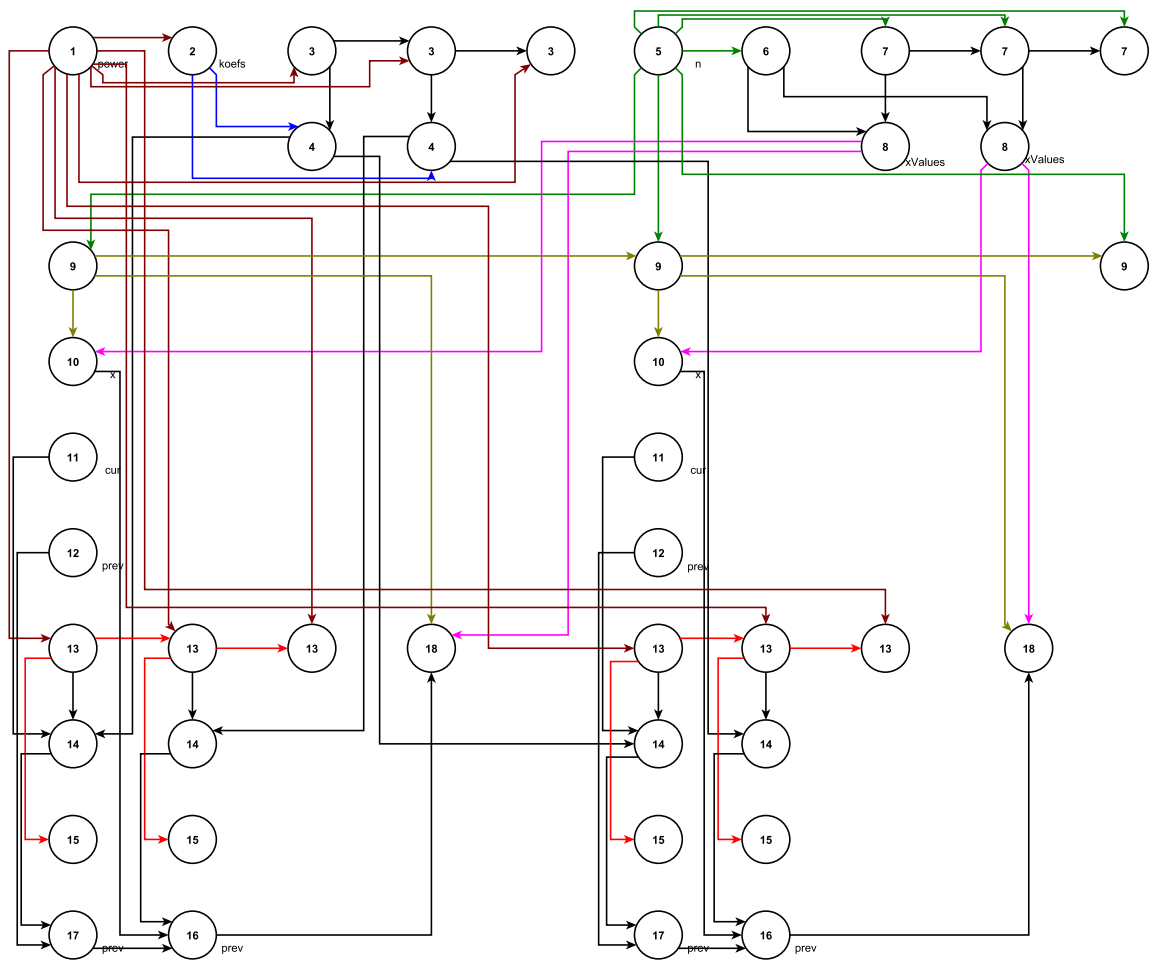


Рисунок 1.4 – Информационная история

Возможность распараллеливания

Один из способов — разделить вычисления на несколько частей, каждую из которых будет обрабатывать свой поток. Например, можно разделить массив коэффициентов на равные подмассивы и запустить вычисление частей полинома в отдельных потоках. То есть каждый поток будет обрабатывать новый подмассив как отдельный полином.