



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»

Домашнее задание
по курсу «Анализ Алгоритмов»

Тема Графовые модели программ

Студент Фам Минь Хиеу

Группа ИУ7-52Б

Оценка (баллы) _____

Преподаватель Волкова Л. Л.

Москва — 2024 г.

Содержание

1	Технологическая часть	3
1.1	Исходный код программы	3
1.2	Модели графов	4
	Заключение	7

1 Технологическая часть

Вариант 4 лабораторной работы заключается в реализации алгоритма поиска подстроки в строки КМП, используя параллельные вычисления. Для домашнего задания выбран алгоритм получения массива `lps` (longest prefix suffix). При выполнении домашнего задания использовался язык программирования C++. Необходимо реализовать граф управления программы, информационный граф программы, операционную историю программы, информационную историю программы.

1.1 Исходный код программы

На листинке 1.1 приведен код программы, которая формирует матрицу без i -й строки и j -го столбца.

Листинг 1.1 — Заполнение массива `lps`

```
1 void computeLPSArray(string pattern, vector<int>& lps)
2 {
3     int n = lps.size(); //1
4     int len = 0; //2
5     int i = 1; //3
6     cout << "start cycle" << endl; //4
7     while (i < n) //5
8     {
9         if (pattern[i] == pattern[len]) //6
10        {
11            len++; //7
12            lps[i] = len; //8
13            i++; //9
14        }
15        else //10
16        {
17            if (len != 0) //11
18                len = lps[len - 1]; //12
19            else //13
20            {
21                lps[i] = len; //14
22                i++; //15
23            }
24        }
25    }
26 }
```

1.2 Модели графов

Вершины А и В связываются операционным отношением, если вершина В может быть выполнена сразу после А. Вершины А и В связаны информационным отношением, если вершина В использует результат работы вершины А. Для операционного графа вершинами являются операторы, а дугами операционные отношения. Для информационного графа вершинами являются операторы, а дугами информационные отношения. Для графа операционной истории вершинами является срабатывание операторов, а дугами операционные отношения. Для графа информационной истории вершинами являются срабатывания операторов, а дугами информационные отношения. На рисунках 1.1–1.4 представлены граф управления программы, информационный граф программы, операционная история программы, информационная история программы.

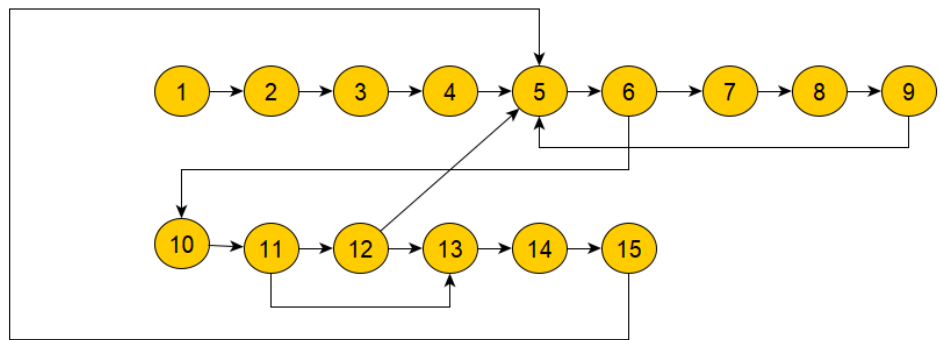


Рисунок 1.1 — Граф управления программы

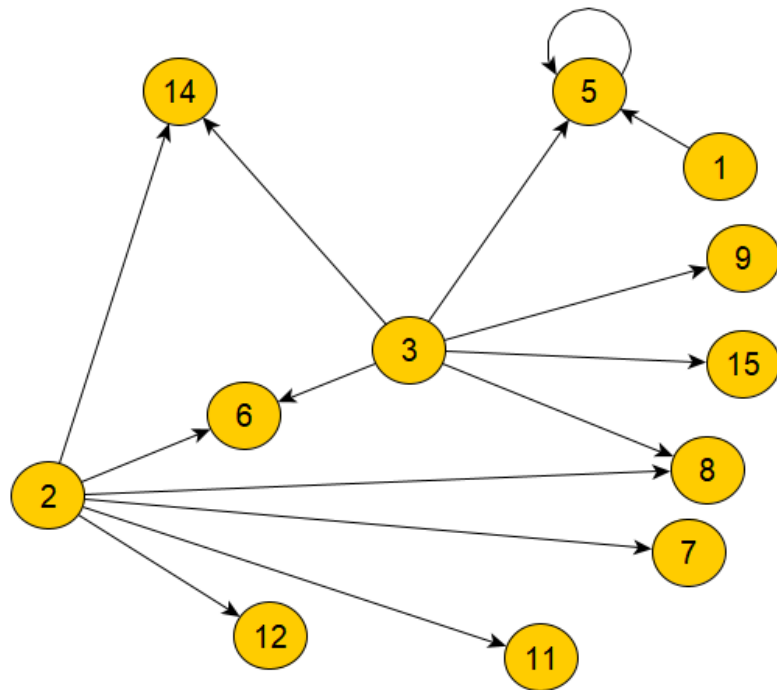


Рисунок 1.2 — Информационный граф программы

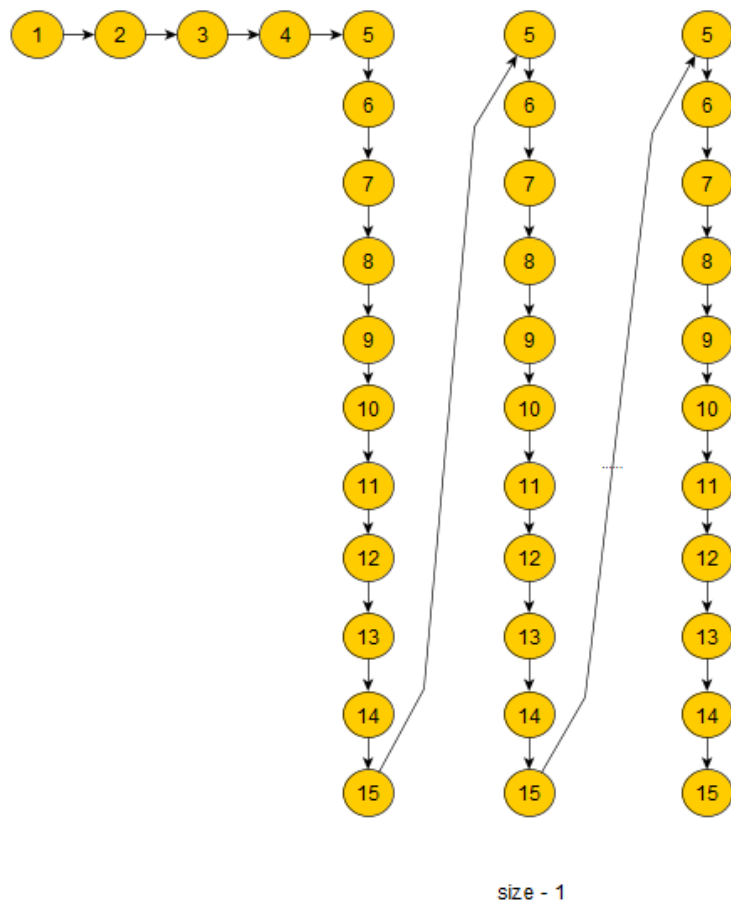


Рисунок 1.3 — Операционная история программы

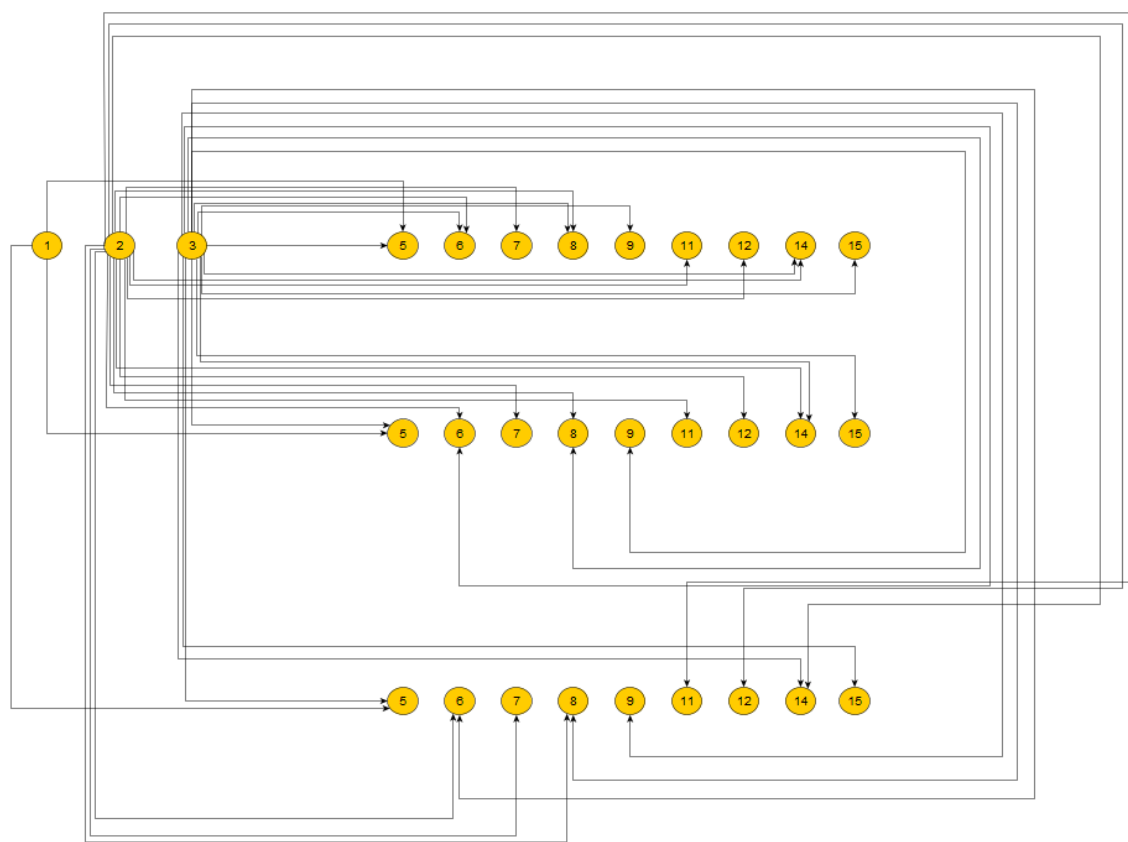


Рисунок 1.4 — Информационная история

Заключение

Графовые модели были успешно разработаны.