Министерство Науки и Образования Российской Федерации

Санкт-Петербургский Государственный Электротехнический Университет «ЛЭТИ»

Кафедра МО ЭВМ

Отчёт по лабораторной работе №1-2

**«**Построение операционных графовых моделей последовательных

программ и их анализ на основе аппарата

поглощающих цепей Маркова**»**

**Вариант 19.**

**Выполнил: Эмман П.А.**

**гр. 3351**

**Проверил: Кирьянчиков В.А.**

**Санкт-Петербург**

**2007**

Оглавление

[Задание 3](#_Toc186207852)

[Ход работы 4](#_Toc186207853)

[Программа для анализа 4](#_Toc186207854)

[Граф программы 6](#_Toc186207855)

[Cтруктурирование графа для метода поуровневой детализации: 11](#_Toc186207856)

[Результаты вычисления матриц при помощи fm.exe 12](#_Toc186207857)

[На входе 12](#_Toc186207858)

[Поуровневая детализация 12](#_Toc186207859)

[Вычисление средних и дисперсий по вычисленной ФМ 14](#_Toc186207860)

[Вывод 16](#_Toc186207861)

# Задание

# Для рассматривавшегося в лабораторных работах курса «Метрология программного обеспечения» индивидуального задания разработать операционную модель управляющего графа программы на основе схемы алгоритма и ассемблерного представления программы.

## При выполнении работы рекомендуется для упрощения обработки графа и ПЦМ исключить диалог при выполнении операций ввода-вывода данных, а также привести программу к структурированному виду.

# Выбрать вариант графа с нагруженными вершинами, каждая из которых должна представлять фрагмент программы, соответствующий линейному участку или ветвлению.

## При расчете вероятностей ветвлений, зависящих от распределения данных, принять равномерное распределение обрабатываемых данных в ограниченном диапазоне (например, [0,100] - для положительных чисел или [-100,100] - для произвольных чисел).

## В случае ветвлений, вызванных проверкой выхода из цикла, вероятности рассчитываются исходя из априорных сведений о числе повторений цикла.

# В качестве параметров, характеризующих потребление ресурсов, использовать времена выполнения команд соответствующих участков программы, задаваемые в тактах процессора.

# Выполнить оценку времен выполнения каждого линейного участка и каждого ветвления в графе программы. Оценку времен выполнения участков производить либо с использованием монитора (например, Sampler или Vtune), либо прямым подсчетом по тексту программы.

# Ход работы

Особенность данной программы то, что она по сути линейна. При «правильном» ее использовании, вычисление будут о(1). Для проведения измерений мы ее «зацикливаем». Но количество повторов всегда заранее известно: сколько значений будет перебираться до 1,5, и сколько после – по этому соотношению будет соотношение выбора ветки 5-6 5-7. ветка 3-4 представляет собой лишь «страховой случай».

Измененная программа в ходе лабораторных работ по курсу «МПО»:

#include "SAMPLER.H"

#include <math.h>

#include <conio.h>

#include <stdio.h>

#include <stdlib.h>

double x,er,ec;

int done;

const double sqrtpi = 1.7724538;

const double t2 = 0.66666667;

const double t3 = 0.66666667;

const double t4 = 0.07619048;

const double t5 = 0.01693122;

const double t6 = 3.078403E-3;

const double t7 = 4.736005E-4;

const double t8 = 6.314673E-5;

const double t9 = 7.429027E-6;

const double t10 = 7.820028E-7;

const double t11 = 7.447646E-8;

const double t12 = 6.476214E-9;

double erf(double x)

{

double x2 = x \* x;

double sum = t5+x2\*(t6+x2\*(t7+x2\*(t8+x2\*(t9+x2\*(t10+x2\*(t11+x2\*t12))))));

return 2.0\*exp(-x2)/sqrtpi\*(x\*(1+x2\*(t2+x2\*(t3+x2\*(t4+x2\*sum)))));

}// { function erf }

double erfc(double x)

{

double x2 = x \* x;

double v =1.0/(2.0\*x2);

double sum = v/(1+8\*v/(1+9\*v/(1+10\*v/(1+11\*v/(1+12\*v)))));

sum =v/(1+3\*v/(1+4\*v/(1+5\*v/(1+6\*v/(1+7\*sum)))));

return (1.0/(exp(x2)\*x\*sqrtpi\*(1+v/(1+2\*sum))));

}// { function ercf }

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вершина | Код sampler | проходов | Время | вес | Текст программы |
| 1 | 42-46 | 1 | 5,03 | 5,03 | x = 0.00f;  double m = (rand() % 18) / 10 + 1.6; |
| 2 | 46-49  75-49  75-77 | 1  520  1 | **0,01**  0,51  0,84 | 0,5097 | while(x < m){ |
| 3 | 49-52  49-58 | 1  520 | 0,84  0,42 | 0,4208 | if (x == 0.0) |
| 4 | 52-55 | 1 | 0,84 | 0,84 | {  er = 0.0;  ec = 1.0;  } |
|  |  |  |  |  | else  { |
| 5 | 58-61  58-67 | 300  220 | 1,06  0,37 | 0,768 | if (x<1.5) |
| 6 | 61-64 | 300 | 0,56 | 0,56 | {  er = erf(x);  ec = 1.0 - er;  } |
|  |  |  |  |  | else |
| 7 | 67-70 | 220 | 0,75 | 0,75 | {  ec = erfc(x);  er = 1.0 - ec;  } |
|  |  |  |  |  | } |
| 8 | 55-73  64-73  70-73 | 1  300  220 | **0,01**  0,37  0,69 | 0,5044 | x += 0.01; |
| 9 |  |  |  |  | }  } |

void main(void)

{

SAMPLE; //42

x = 0.00f;

randomize();

double m = (rand() % 18) / 10 + 1.6;

SAMPLE; //46

while(x < m){

SAMPLE;//49

if (x == 0.0)

{

SAMPLE;//52

er = 0.0;

ec = 1.0;

SAMPLE;//55

}else

{

SAMPLE;//58

if (x<1.5)

{

SAMPLE;//61

er = erf(x);

ec = 1.0 - er;

SAMPLE;//64

}else

{

SAMPLE;//67

ec = erfc(x);

er = 1.0 - ec;

SAMPLE;//70

}

}

SAMPLE;//73

x += 0.005;

SAMPLE;//75

}

SAMPLE;//77

}

# Граф программы



#### Проверим это по семплеру:

----------------------------------------------------------------------

Исх.Поз. Прием.Поз. Общее время(мкс) Кол-во прох. Среднее время(мкс)

----------------------------------------------------------------------

1 : 42 1 : 46 5.03 1 5.03

----------------------------------------------------------------------

1 : 46 1 : 49 0.00 1 0.00

----------------------------------------------------------------------

1 : 49 1 : 52 0.84 1 0.84

1 : 49 1 : 58 218.74 520 0.42

----------------------------------------------------------------------

1 : 52 1 : 55 0.84 1 0.84

----------------------------------------------------------------------

1 : 55 1 : 73 0.00 1 0.00

----------------------------------------------------------------------

1 : 58 1 : 61 318.48 300 1.06

1 : 58 1 : 67 80.46 220 0.37

----------------------------------------------------------------------

1 : 61 1 : 64 169.30 300 0.56

----------------------------------------------------------------------

1 : 64 1 : 73 110.63 300 0.37

----------------------------------------------------------------------

1 : 67 1 : 70 165.94 220 0.75

----------------------------------------------------------------------

1 : 70 1 : 73 151.70 220 0.69

----------------------------------------------------------------------

1 : 73 1 : 75 416.53 521 0.80

----------------------------------------------------------------------

1 : 75 1 : 49 267.35 520 0.51

1 : 75 1 : 77 0.84 1 0.84

----------------------------------------------------------------------

# Для полученного графа построить соответствующую ему поглощающую цепь Маркова (ПЦМ), определить ее фундаментальную матрицу(ФМ) и вектор нагрузочных парметров L.

# Вычислить оценки средних времен, дисперсии и СКО времен выполнения как всей программы, так и ее основных фрагментов, на которые она может быть разбита.

# Определение ФМ ПЦМ требуется выполнить двумя способами:

# путем непосредственного обращения матрицы (I-Q), полученной по переходной матрице ПЦМ, соответствующей графу всей программы;

# путем структурной детализации фундаментальных матриц, соответствующих подграфам элементарных вычислительных процессов.

# Оценки времен выполнения следует определить как в тактах, так и в абсолютных единицах времени (сек, мсек или мксек). Результаты расчетов представить в виде таблиц как для всей программы, так и для ее фрагментов.

**ВЕКТОР ЗАТРАТ**



|  |  |
| --- | --- |
| ВЕРШИНА | ВЕС |
| 1 | 5,03 |
| 2 | 0,5097 |
| 3 | 0,4208 |
| 4 | 0,84 |
| 5 | 0,768 |
| 6 | 0,56 |
| 7 | 0,75 |
| 8 | 0,504 |
| 9 | 0,01 |

|  |  |  |
| --- | --- | --- |
| ВЕРШИНА | ПЕРЕХОД | ВЕРОЯТНОСТЬ |
| 2 | 2-9 | 1/521 = 0,0019 |
| 2-3 | 520/521 = 0,9981 |
| 3 | 3-4 | 1/521 = 0,0019 |
| 3-5 | 520/521 = 0,9981 |
| 5 | 5-6 | 300/520 = 0,5769 |
| 5-7 | 220/520 = 0,4231 |

## Cтруктурирование графа для метода поуровневой детализации:

**S3**



|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

**S2**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 1/(1-q) | Q/(1-q) | 1 |
| 0 | 1/(1-q) | Q/(1-q) | 1 |
| 0 | 1/(1-q) | 1/(1-q) | 1 |
| 0 | 0 | 0 | 1 |

Q = 0,9981

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 526,3157 | 525,3158 | 1 |
| 0 | 526,3157 | 525,3158 | 1 |
| 0 | 526,3157 | 526,3157 | 1 |
| 0 | 0 | 0 | 1 |

**S1**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | p | 1-p | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

p = 0,5769

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0,5769 | 0,4231 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

## Результаты вычисления матриц при помощи fm.exe

## На входе

tops {

C1(5.03),

C2(0.5097),

B1(0.4208),

B2(0.84),

A1(0.768),

A2(0.56),

A3(0.75),

A4(0.01),

B3(0.504),

C3(0.01)

}

links {

C1->C2(1),

C2->B1(0.9981),

B1->B2(0.0019),

B1->A1(0.9981),

A1->A2(0.5769),

A1->A3(0.4231),

A2->A4(1),

A3->A4(1),

A4->B3(1),

B2->B3(1),

B3->C2(1),

C2->C3(0.0019)

}

## Поуровневая детализация

Файл получен с помощью программы fm.exe Version 1.0.1 (build Dec 14 2000 17:32:35)

Вызвано распознавание шаблона для подграфа @ROOT:

начальная вершина C1, конечная вершина C3

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@ROOT ║ 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Конструкция ЦИКЛ

Вершина проверки условия: C2

Дуга, входящая в тело цикла: C2->B1

Дуга, выходящая из тела цикла: B3->C2

Вероятность повторения тела цикла: 0.9981

Вероятность выхода из цикла: 0.0019

Матрица для подстановки:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 1 ║

C2 ║ 0 526.3 525.3 1 ║

@TMP0 ║ 0 526.3 526.3 1 ║

C3 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Вызвано распознавание шаблона для подграфа @TMP0:

начальная вершина B1, конечная вершина B3

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 1 ║

C2 ║ 0 526.3 525.3 1 ║

@TMP0 ║ 0 526.3 526.3 1 ║

C3 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Конструкция ВЕТВЛЕНИЕ

Матрица для подстановки:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

B1 ║ 1 0.9981 0.0019 1 ║

@TMP1 ║ 0 1 0 1 ║

@TMP2 ║ 0 0 1 1 ║

B3 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Вызвано распознавание шаблона для подграфа @TMP1:

начальная вершина A1, конечная вершина A4

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 524.3 0.9981 525.3 1 ║

C2 ║ 0 526.3 525.3 524.3 0.9981 525.3 1 ║

B1 ║ 0 526.3 526.3 525.3 1 526.3 1 ║

@TMP1 ║ 0 526.3 525.3 525.3 0.9981 526.3 1 ║

@TMP2 ║ 0 526.3 525.3 524.3 1.998 526.3 1 ║

B3 ║ 0 526.3 525.3 524.3 0.9981 526.3 1 ║

C3 ║ 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Конструкция ВЕТВЛЕНИЕ

Матрица для подстановки:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.4231 0.5769 1 ║

@TMP3 ║ 0 1 0 1 ║

@TMP4 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Вызвано распознавание шаблона для подграфа @TMP3:

начальная вершина A3, конечная вершина A3

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

C2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

B1 ║ 0 526.3 526.3 525.3 222.3 303.1 525.3 1 526.3 1 ║

A1 ║ 0 526.3 525.3 525.3 222.3 303.1 525.3 0.9981 526.3 1 ║

@TMP3 ║ 0 526.3 525.3 524.3 222.8 302.5 525.3 0.9981 526.3 1 ║

@TMP4 ║ 0 526.3 525.3 524.3 221.8 303.5 525.3 0.9981 526.3 1 ║

A4 ║ 0 526.3 525.3 524.3 221.8 302.5 525.3 0.9981 526.3 1 ║

@TMP2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 1.998 526.3 1 ║

B3 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Шаблон распознан как атомарная операция

Вызвано распознавание шаблона для подграфа @TMP4:

начальная вершина A2, конечная вершина A2

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

C2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

B1 ║ 0 526.3 526.3 525.3 222.3 303.1 525.3 1 526.3 1 ║

A1 ║ 0 526.3 525.3 525.3 222.3 303.1 525.3 0.9981 526.3 1 ║

A3 ║ 0 526.3 525.3 524.3 222.8 302.5 525.3 0.9981 526.3 1 ║

@TMP4 ║ 0 526.3 525.3 524.3 221.8 303.5 525.3 0.9981 526.3 1 ║

A4 ║ 0 526.3 525.3 524.3 221.8 302.5 525.3 0.9981 526.3 1 ║

@TMP2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 1.998 526.3 1 ║

B3 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Шаблон распознан как атомарная операция

Вызвано распознавание шаблона для подграфа @TMP2:

начальная вершина B2, конечная вершина B2

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

C2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

B1 ║ 0 526.3 526.3 525.3 222.3 303.1 525.3 1 526.3 1 ║

A1 ║ 0 526.3 525.3 525.3 222.3 303.1 525.3 0.9981 526.3 1 ║

A3 ║ 0 526.3 525.3 524.3 222.8 302.5 525.3 0.9981 526.3 1 ║

A2 ║ 0 526.3 525.3 524.3 221.8 303.5 525.3 0.9981 526.3 1 ║

A4 ║ 0 526.3 525.3 524.3 221.8 302.5 525.3 0.9981 526.3 1 ║

@TMP2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 1.998 526.3 1 ║

B3 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Шаблон распознан как атомарная операция

Результат:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

C2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 525.3 1 ║

B1 ║ 0 526.3 526.3 525.3 222.3 303.1 525.3 1 526.3 1 ║

A1 ║ 0 526.3 525.3 525.3 222.3 303.1 525.3 0.9981 526.3 1 ║

A3 ║ 0 526.3 525.3 524.3 222.8 302.5 525.3 0.9981 526.3 1 ║

A2 ║ 0 526.3 525.3 524.3 221.8 303.5 525.3 0.9981 526.3 1 ║

A4 ║ 0 526.3 525.3 524.3 221.8 302.5 525.3 0.9981 526.3 1 ║

B2 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 1.998 526.3 1 ║

B3 ║ 0 526.3 525.3 524.3 221.8 302.5 524.3 0.9981 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Алгебраический метод**

Результат:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 525.3 1 ║

C2 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 525.3 1 ║

B1 ║ 0 526.3 526.3 1 525.3 303.1 222.3 525.3 526.3 1 ║

B2 ║ 0 526.3 525.3 1.998 524.3 302.5 221.8 524.3 526.3 1 ║

A1 ║ 0 526.3 525.3 0.9981 525.3 303.1 222.3 525.3 526.3 1 ║

A2 ║ 0 526.3 525.3 0.9981 524.3 303.5 221.8 525.3 526.3 1 ║

A3 ║ 0 526.3 525.3 0.9981 524.3 302.5 222.8 525.3 526.3 1 ║

A4 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 525.3 526.3 1 ║

B3 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Вычисление средних и дисперсий по вычисленной ФМ

Файл получен с помощью программы fm.exe Version 1.0.1 (build Dec 14 2000 17:32:35)

Фундаментальная матрица:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 525.3 1 ║

C2 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 525.3 1 ║

B1 ║ 0 526.3 526.3 1 525.3 303.1 222.3 525.3 526.3 1 ║

B2 ║ 0 526.3 525.3 1.998 524.3 302.5 221.8 524.3 526.3 1 ║

A1 ║ 0 526.3 525.3 0.9981 525.3 303.1 222.3 525.3 526.3 1 ║

A2 ║ 0 526.3 525.3 0.9981 524.3 303.5 221.8 525.3 526.3 1 ║

A3 ║ 0 526.3 525.3 0.9981 524.3 302.5 222.8 525.3 526.3 1 ║

A4 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 525.3 526.3 1 ║

B3 ║ 0 526.3 525.3 0.9981 524.3 302.5 221.8 524.3 526.3 1 ║

C3 ║ 0 0 0 0 0 0 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Потребление ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 5.03 ║

C2 ║ 0.5097 ║

B1 ║ 0.4208 ║

B2 ║ 0.84 ║

A1 ║ 0.768 ║

A2 ║ 0.56 ║

A3 ║ 0.75 ║

A4 ║ 0.01 ║

B3 ║ 0.504 ║

C3 ║ 0.01 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Средние значения потребления ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 1504 ║

C2 ║ 1499 ║

B1 ║ 1501 ║

B2 ║ 1500 ║

A1 ║ 1501 ║

A2 ║ 1500 ║

A3 ║ 1500 ║

A4 ║ 1499 ║

B3 ║ 1499 ║

C3 ║ 0.01 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Дисперсии потребления ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C1 ║ 2.249e+006 ║

C2 ║ 2.249e+006 ║

B1 ║ 2.249e+006 ║

B2 ║ 2.249e+006 ║

A1 ║ 2.249e+006 ║

A2 ║ 2.249e+006 ║

A3 ║ 2.249e+006 ║

A4 ║ 2.249e+006 ║

B3 ║ 2.249e+006 ║

C3 ║ 0 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Для фрагмента

tops {

A1(0.768),

A2(0.56),

A3(0.75),

A4(0.01)

}

links {

A1->A2(0.5769),

A1->A3(0.4231),

A2->A4(1),

A3->A4(1),

A4->B3(1)

}

#### Поуровневая детализация

Файл получен с помощью программы fm.exe Version 1.0.1 (build Dec 14 2000 17:32:35)

Вызвано распознавание шаблона для подграфа @ROOT:

начальная вершина A1, конечная вершина A4

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@ROOT ║ 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Конструкция ВЕТВЛЕНИЕ

Матрица для подстановки:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.4231 0.5769 1 ║

@TMP0 ║ 0 1 0 1 ║

@TMP1 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Вызвано распознавание шаблона для подграфа @TMP0:

начальная вершина A3, конечная вершина A3

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.4231 0.5769 1 ║

@TMP0 ║ 0 1 0 1 ║

@TMP1 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Шаблон распознан как атомарная операция

Вызвано распознавание шаблона для подграфа @TMP1:

начальная вершина A2, конечная вершина A2

Текущее состояние фундаментальной матрицы:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.4231 0.5769 1 ║

A3 ║ 0 1 0 1 ║

@TMP1 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Шаблон распознан как атомарная операция

Результат:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.4231 0.5769 1 ║

A3 ║ 0 1 0 1 ║

A2 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### Алгебраический метод

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.5769 0.4231 1 ║

A2 ║ 0 1 0 1 ║

A3 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### Средние, дисперсия и ресурсы

Фундаментальная матрица:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1 0.5769 0.4231 1 ║

A2 ║ 0 1 0 1 ║

A3 ║ 0 0 1 1 ║

A4 ║ 0 0 0 1 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Потребление ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 0.768 ║

A2 ║ 0.56 ║

A3 ║ 0.75 ║

A4 ║ 0.01 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Средние значения потребления ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 1.418 ║

A2 ║ 0.57 ║

A3 ║ 0.76 ║

A4 ║ 0.01 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Дисперсии потребления ресурсов:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A1 ║ 0.008812 ║

A2 ║ -5.551e-017 ║

A3 ║ 1.11e-016 ║

A4 ║ 0 ║

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Вывод

В ходе лабораторной работы построен управляющий граф программы, определены вероятности выполнения ветвей графа, потребление процессорного времени участками программы.

При помощи ПС fm.exe построена фундаментальная матрица для заданной программы — двумя способами: алгебраическим (преобразованием матрицы переходов и обращением результата) и методом поуровневой детализации. Вычислены средние знания потребления процессорного времени операциями программы в тактах и микросекундах.