

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №6

по дисциплине «Качество и метрология программного обеспечения»

**ТЕМА: «Оценка характеристик надежности программ по структурным
схемам надежности»**

Студентка гр. 8304

Николаева М. А.

Преподаватель

Кирияничиков В. А.

Санкт-Петербург

2022

Цель работы.

Выполнить расчет характеристик надежности вычислительной системы по структурной схеме надежности, выбранной из таблицы в соответствии с номером студента в списке группы.

Задание.

В качестве оцениваемых характеристик следует рассматривать:

- 1) Вероятность безотказной работы системы в заданный момент времени;
- 2) Среднее время до отказа системы.

Выполнение расчетов следует производить двумя способами:

- 1) Расчетным способом;
- 2) Программным способом с помощью Анализатора структурных схем надежности RSSA (Reliability Structural Scheme Analyzer).

Для реализации расчетного способа следует получить аналитические выражения, позволяющие вычислить требуемые характеристики системы через характеристики надежности ее компонентов. Параметры надежности компонентов системы задаются в виде интенсивностей отказов λ_i , которые считаются постоянными и не зависят от времени. Кроме того, события, заключающиеся в отказе отдельных компонентов системы, следует считать независимыми.

При вычислении интегралов результаты следует получать аналитическим способом, а не с применением стандартных пакетов программ.

Интегралы следует вычислять в общем виде и привести выражения для вычисления МТТФ. И только затем подставлять числа – иначе проверить невозможно.

Для реализации программного способа следует:

- 1) Описать заданную структурную схему надежности (ССН) системы на XML в следующем виде:

```
<Schema>><!--Начало описания схемы-->
  <graf><!--Начало описания блоков-->
    <Block><!--Начало описания текущего блока-->
      <Id>1</Id><!-- id блока, уникально-->
      <Id2>1</Id2><!--id блока, используется для задания совпадающих блоков при
смешанной конфигурации-->
```

```

    <failureRate>0</failureRate><!--Интенсивность отказов-->
    <name>1</name><!--Имя блока-->
    <quantity>1</quantity><!--количество последовательно соединенных одинаковых
блоков-->

    <list><!--список блоков, исходящих из данного-->
        <int>2</int>
        <int>3</int>
    </list>
    <type></type><!--тип, простой (по умолчанию) или составной (пока не используется)-
->

</Block><!--Конец описания текущего блока-->

<Block>
    <Id>2</Id>
    <Id2>2</Id2>
    <failureRate>2.28E-5</failureRate>
    <name>2</name>
    <quantity>1</quantity>
    <list>
        <int>5</int>
    </list>
    <type></type>
</Block>
</graf><!--Конец описания блоков-->
<ListOfFlag/><!--Служебные строки завершения-->
<listOfNode/><!--описания схемы, в таком виде -->
<list/><!-- должны использоваться для всех вариантов-->
</Schema><!--Конец описания схемы-->

```

Примечание: подготовленный файл сохранить в формате xml.

- 2) Установить Java – машину версии 8 (при ее отсутствии – скачать из Интернета).
- 3) Скопировать каталог rssa в свой рабочий каталог.
- 4) Запустить программу Анализатор ССН с помощью файлов, указанных в файле readme.txt.
- 5) В главном окне программы с помощью меню выбрать файл с подготовленным описанием ССН и вычислить требуемые характеристики надежности (см. файл Описание анализатора ССН, содержащий краткое руководство пользователя).

- б) Полученные с помощью программы RSSA значения характеристик надежности системы следует сравнить с характеристиками, вычисленными расчетным способом. В случае существенного отличия характеристик разобраться в причинах их расхождения.

Ход работы.

По списку был выбран вариант № 13. Описание варианта представлено в таблице 1.

Таблица 1 – Исходные данные.

N_1					N_2		N_3	
комбинат. соединения	λ_1	λ_2	λ_3	λ_4	комб. соедин.	λ	комб. соедин.	λ
C(3)	2.8	4.0	1.8	-	(3,2)	1.8	(1,1)	2.2

Структура вычислительной системы представляет собой три блока: N_1 - состоит из трех последовательных блоков, N_2 - из двух параллельных ветвей (3 блока на верхней ветви, 2 на нижней), N_3 - из двух параллельных ветвей (1 блок на верхней ветви, 1 на нижней).

Был построен граф программы с двумя мнимыми вершинами для перехода от N_2 к N_3 и для создания конечной вершины. Результат представлен на рис. 1.

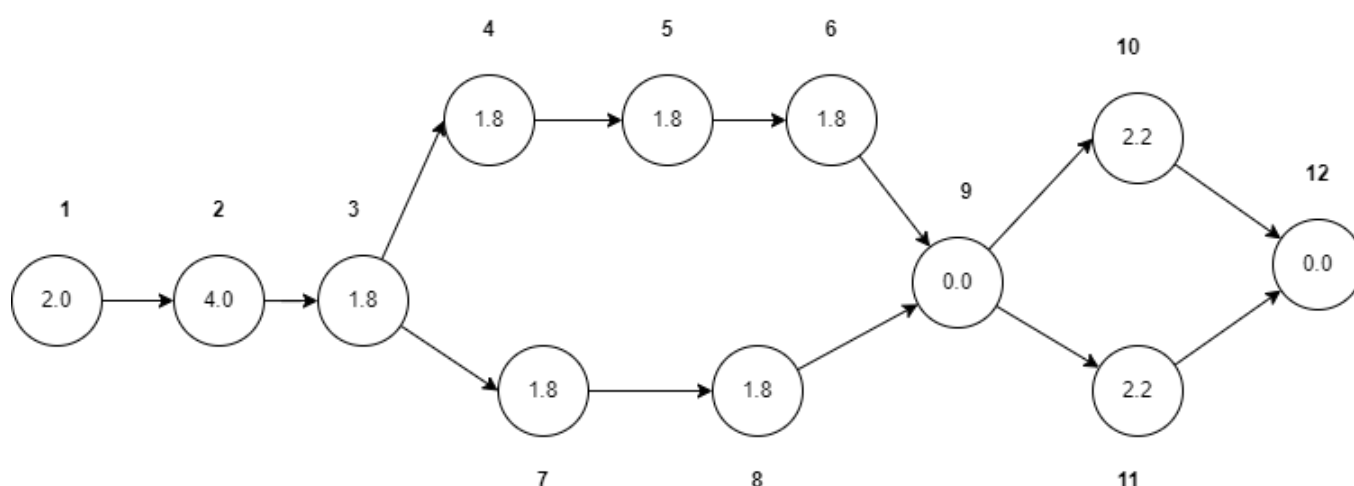


Рисунок 1 - Граф программы

1) Ручной расчет.

Был выполнен ручной расчет вероятностей для участков и для целого графа, а также среднее время работы до отказа для значения $t = 2$.

$$R_{N_1} = e^{-(\lambda_1 + \lambda_2 + \lambda_3)t} = e^{-(2.8 + 4.0 + 1.8) * 2 * 10^{-5}} = 0.9998280148;$$

$$R_{N_2} = 1 - (1 - e^{-3\lambda_{4-6}t})(1 - e^{-2\lambda_{7-8}t}) = 1 - (1 - e^{-3 * 1.8 * 2 * 10^{-5}})(1 - e^{-2 * 1.8 * 2 * 10^{-5}}) \\ = 0.9999999922 \cong 1;$$

$$R_{N_3} = 1 - (1 - e^{-\lambda_{10-11}t})^2 = 1 - (1 - e^{-2.2 * 2 * 10^{-5}})^2 = 0.9999999922 \cong 1;$$

$$R_S = R_{N_1} * R_{N_2} * R_{N_3} \approx 0.999828005;$$

$$MTTF = \int_0^{\infty} R_S(t) dt = \\ = \int_0^{\infty} e^{-(\lambda_1 + \lambda_2 + \lambda_3)t} * \left(1 - (1 - e^{-3\lambda_{4-6}t}) * (1 - e^{-2\lambda_{7-8}t})\right) * \\ * \left(1 - (1 - e^{-\lambda_{10-11}t})^2\right) dt = 9218.23135248.$$

В результате вычислений получили, что вероятность безотказной работы системы в заданный момент времени равна 0.999828005, а среднее время до отказа системы 9218.23135248 часа.

2) Программный расчет.

Был выполнен программный расчет. XML-описание графа представлено в приложении А. Построенная схема представлена на рисунке 2. Схема соответствует изначально построенному графу.

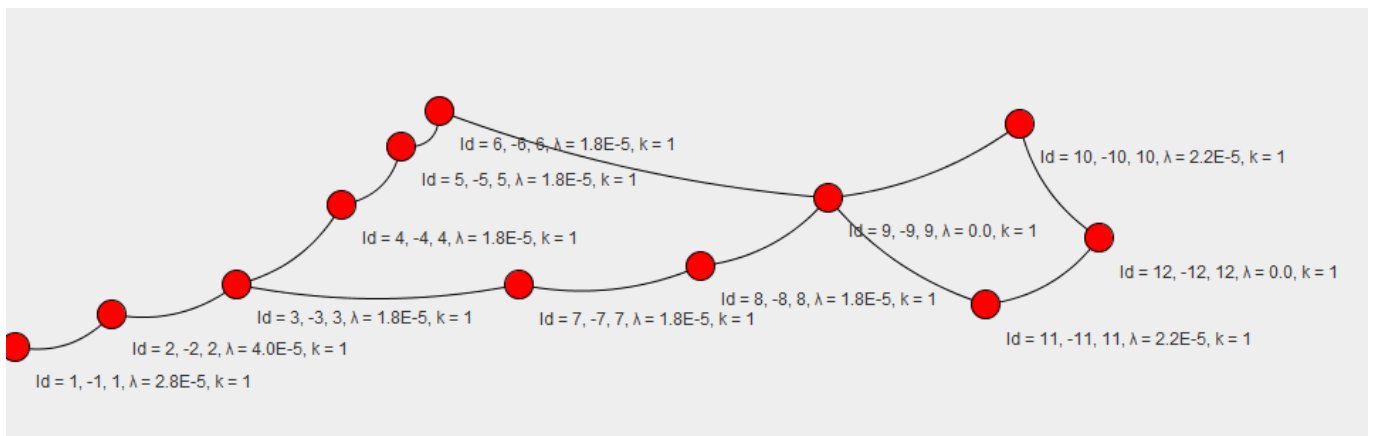


Рисунок 2 – Программное построение графа

Результаты вычисления надежности и среднего времени безотказной работы представлены на рисунке 3.

t	R	T
2.0	0.9998280050816071	9213.170834581513

Рисунок 3 – Программный расчет надежности и времени

Выводы.

В ходе выполнения лабораторной работы было проведено исследование характеристик надежности вычислительных систем по структурным схемам надежности. По данной структуре были рассчитаны вероятность безотказной работы системы в заданный момент времени и среднее время до отказа системы двумя способами (ручным и программным).

Общая надежность системы при ручном расчёте полностью совпадает с общей надежностью, рассчитанной программным способом при округлении до девяти знаков после запятой. Среднее время до отказа системы при ручном расчёте практически совпадает со средним временем до отказа системы, рассчитанным программным способом. Разницу можно объяснить тем, что программные вычисления являются более точными.

ПРИЛОЖЕНИЕ А.

XML-ОПИСАНИЕ.

```
<Schema>
  <graf>
    <Block>
      <Id>1</Id>
      <Id2>1</Id2>
      <failureRate>2.8E-5</failureRate>
      <name>1</name>
      <quantity>1</quantity>
      <list>
        <int>2</int>
      </list>
      <type></type>
    </Block>

    <Block>
      <Id>2</Id>
      <Id2>2</Id2>
      <failureRate>4.0E-5</failureRate>
      <name>2</name>
      <quantity>1</quantity>
      <list>
        <int>3</int>
      </list>
      <type></type>
    </Block>

    <Block>
      <Id>3</Id>
      <Id2>3</Id2>
      <failureRate>1.8E-5</failureRate>
      <name>3</name>
      <quantity>1</quantity>
      <list>
        <int>4</int>
        <int>7</int>
      </list>
      <type></type>
    </Block>

    <Block>
      <Id>4</Id>
      <Id2>4</Id2>
      <failureRate>1.8E-5</failureRate>
      <name>4</name>
```

```

    <quantity>1</quantity>
    <list>
      <int>5</int>
    </list>
    <type></type>
  </Block>

  <Block>
    <Id>5</Id>
    <Id2>5</Id2>
    <failureRate>1.8E-5</failureRate>
    <name>5</name>
    <quantity>1</quantity>
    <list>
      <int>6</int>
    </list>
    <type></type>
  </Block>

  <Block>
    <Id>6</Id>
    <Id2>6</Id2>
    <failureRate>1.8E-5</failureRate>
    <name>6</name>
    <quantity>1</quantity>
    <list>
      <int>9</int>
    </list>
    <type></type>
  </Block>

  <Block>
    <Id>7</Id>
    <Id2>7</Id2>
    <failureRate>1.8E-5</failureRate>
    <name>7</name>
    <quantity>1</quantity>
    <list>
      <int>8</int>
    </list>
    <type></type>
  </Block>

  <Block>
    <Id>8</Id>
    <Id2>8</Id2>
    <failureRate>1.8E-5</failureRate>
    <name>8</name>

```



```

    <quantity>1</quantity>
    <list>
      <int>9</int>
    </list>
    <type></type>
</Block>

<Block>
  <Id>9</Id>
  <Id2>9</Id2>
  <failureRate>0E-5</failureRate>
  <name>9</name>
  <quantity>1</quantity>
  <list>
    <int>10</int>
    <int>11</int>
  </list>
  <type></type>
</Block>

<Block>
  <Id>10</Id>
  <Id2>10</Id2>
  <failureRate>2.2E-5</failureRate>
  <name>10</name>
  <quantity>1</quantity>
  <list>
    <int>12</int>
  </list>
  <type></type>
</Block>

<Block>
  <Id>11</Id>
  <Id2>11</Id2>
  <failureRate>2.2E-5</failureRate>
  <name>11</name>
  <quantity>1</quantity>
  <list>
    <int>12</int>
  </list>
  <type></type>
</Block>

<Block>
  <Id>12</Id>
  <Id2>12</Id2>
  <failureRate>0E-5</failureRate>

```

```
    <name>12</name>
    <quantity>1</quantity>
    <list></list>
    <type></type>
  </Block>

</graf>
<ListOfFlag />
<listOfNode />
<list />
</Schema>
```