

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Э. БАУМАНА

ФАКУЛЬТЕТ «ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ»

КАФЕДРА «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭВМ

И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»



Моделирование

ЛАБОРАТОРНАЯ РАБОТА №5

МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННОГО ЦЕНТРА

Студент: Петухов И.С.

Группа: ИУ7-71

Преподаватель: Рудаков И.В.

Москва, 2016

Содержание

1	Аналитический раздел	3
2	Технологический раздел	4
2.1	Язык программирования	4
2.2	Примеры кода	4
2.3	Взаимодействие с пользователем	5

1 Аналитический раздел

Цель данной работы - смоделировать работу информационного центра.

В информационный центр приходят клиенты через интервалы времени 10 ± 2 минуты. Если все три имеющихся оператора заняты, то клиенту отказывают в обслуживании. Операторы имеют производительность: 20 ± 5 мин, 40 ± 10 , 40 ± 20 . Клиенты стараются занять свободного оператора с максимальной производительностью. Полученные запросы сдаются в приемный накопитель, откуда выбираются для обработки на первый компьютер для 1 и 2 оператора и на второй компьютер для 3 оператора. Время обработки первого и второго компьютера 15 и 30 мин. Смоделировать процесс обработки для 300 запросов. Определить вероятность отказа. Предусмотреть 300 как на входе, так и на выходе. У компьютеров очереди бесконечные. Оператор освобождается, когда он передаст заявку компьютеру.

Результат При заданных параметрах вероятность отказа клиенту в обслуживании равно 20-25 %.

2 Технологический раздел

2.1 Язык программирования

В качестве языка программирования выбран язык высокого уровня JavaScript.

2.2 Примеры кода

Листинг 2.1 — Программная модель клиента

```
1 function SourceOfInformation(min, max) {
2   let self = this;
3
4   self.min = min;
5   self.max = max;
6
7   let isRequest = false;
8
9   let timeNewOrder = randFromMinToMax(self.min, self.max);
10
11  self.isRequest = function(nowTime) {
12    if (nowTime >= timeNewOrder && !isRequest) {
13      isRequest = true;
14      timeNewOrder += randFromMinToMax(self.min, self.max);
15    }
16
17    return isRequest;
18  }
19
20  self.reset = function() {
21    isRequest = false;
22  }
23 }
```

Листинг 2.2 — Программная модель компьютера

```
1 function Store(time) {
2   let self = this;
3
4   self.time = time;
5
6   let orders = [];
7
8   self.pushOrder = function(nowTime) {
9     orders.push(nowTime);
10  }
11 }
```

```

12 self.run = function(nowTime) {
13     //orders.sort((a, b) => a - b);
14
15     let i = 0;
16     while(orders[i] + self.time < nowTime) {
17         orders.shift();
18         ++i;
19     }
20
21     return i;
22 }
23 }

```

Листинг 2.3 — Программная модель оператора

```

1 function ServiceUnit(min, max, store) {
2     let self = this;
3
4     self.min = min;
5     self.max = max;
6     self.store = store;
7
8     let isFree = false;
9
10    let timeNewWork = randFromMinToMax(self.min, self.max);
11
12    self.isFree = function(nowTime) {
13        if (nowTime >= timeNewWork && !isFree) {
14            isFree = true;
15            self.store.pushOrder(nowTime);
16        }
17
18        return isFree;
19    }
20
21    self.pushWork = function(nowTime) {
22        isFree = false;
23        timeNewWork = nowTime + randFromMinToMax(self.min, self.max);
24    }
25 }

```

2.3 Взаимодействие с пользователем

Взаимодействие с пользователем осуществляется через html страницы, открытые в браузере.

Управляющая программа имитационной модели информационного центра (принцип dt)

Параметры имитационной модели

шаг времени:
кол-во заявок
☐ На входе

Клиенты:

интервал времени прихода:

min:
max:

Операторы:

№1 интервал времени обслуживания:	№2 интервал времени обслуживания:	№3 интервал времени обслуживания:
min: <input type="text" value="15"/>	min: <input type="text" value="30"/>	min: <input type="text" value="20"/>
max: <input type="text" value="25"/>	max: <input type="text" value="50"/>	max: <input type="text" value="60"/>

Приемные накопители:

№1 время обработки:	№2 время обработки:
время: <input type="text" value="15"/>	время: <input type="text" value="30"/>

Рисунок 2.1 — Настройка параметров модели

Запуск

Вероятность отказа: 0.22

Кол-во потерянных заявок: 110

Кол-во зашедших заявок: 500

Кол-во обработанных заявок: 300

Время работы системы: 4757

График отказов в обслуживании

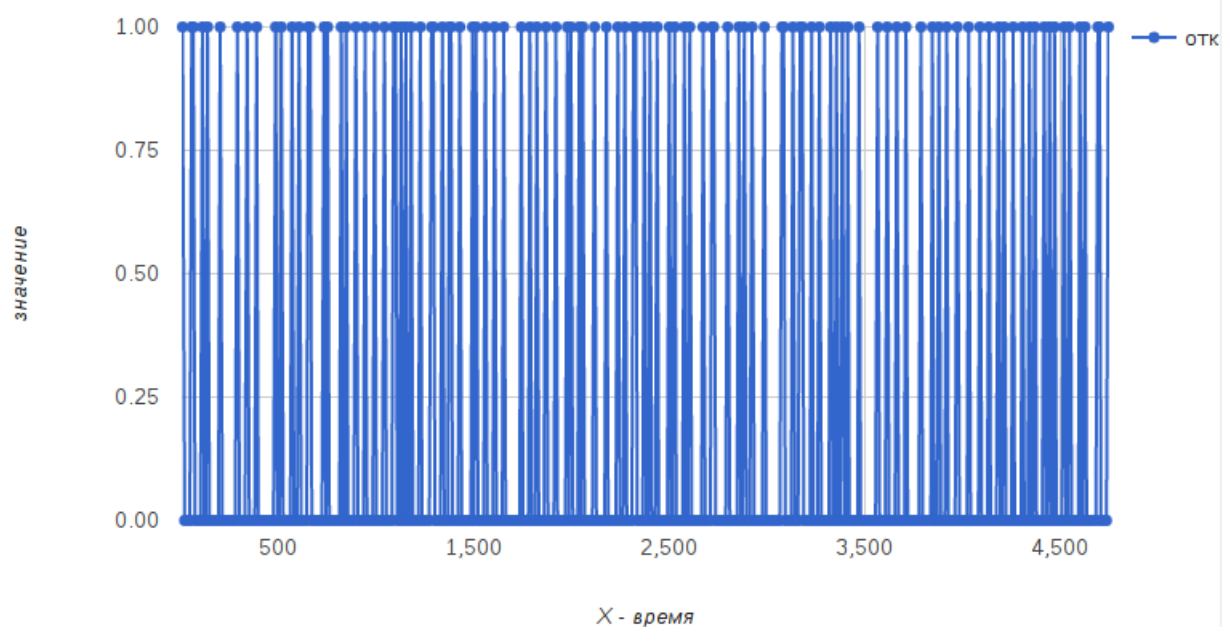


Рисунок 2.2 — Результаты