|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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**

*к лабораторной работе №6*

*По курсу: «Моделирование»*

*Тема:* ***«Моделирование работы автосервиса»***

Студент ИУ7-74Б

Жабин Д.В.

Преподаватель

Рудаков И.В.

*Москва, 2022 г.*

# Задание

В автосервис приезжают клиенты каждые 17 ± 3 минуты. Клиентов обслуживают два менеджера, но, если оба менеджера заняты, клиенту отказывают в обслуживании. Менеджеры имеют разную производительность и могут обслужить клиента за 30 ± 4 и 33 ± 5 минут соответственно. Клиенты стремятся попасть к свободному менеджеру с большей производительностью. После определения менеджером перечня необходимых работ клиенты оказываются в очереди и ждут, пока свободный механик начнет обслуживать их автомобиль. Автомобилями клиентов, обслуженных первым менеджером, занимаются первый, второй и третий механики, а автомобилями клиентов, обслуженных вторым менеджером – четвертый и пятый механики. Механики выполняют свою работу за 14 ± 4, 15 ± 5, 17 ± 2, 18 ± 2 и 21 ± 3 минуты соответственно. Промоделировать процесс обслуживания 1000 клиентов. Определить вероятность отказа.

# Теоретическая часть

В процессе взаимодействия клиентов с автосервисом возможны следующие ситуации:

1) Режим нормального обслуживания, т.е. клиент попадает к одному из свободных менеджеров (более эффективному).

2) Режим отказа в обслуживании клиента, когда все менеджеры заняты.

**Эндогенные переменные**: время обслуживания клиента i-ым менеджером, время работы j-ого механика.

**Экзогенные переменные**: число обслуженных клиентов и число клиентов, получивших отказ.

Структурная схема модели представлена на рисунке 1.

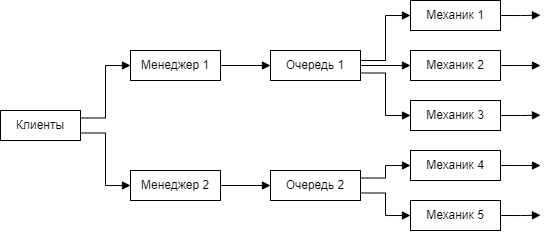


Рисунок 1 – Структурная схема модели

Вероятность отказа вычисляется следующим образом:



# Текст программы

|  |
| --- |
| from random import random  EPS = 1e-5  dt = 0.01  class EvenDistribution:  def \_\_init\_\_(self, a, b):  self.a = a  self.b = b  def generate(self):  return self.a + (self.b - self.a) \* random()  class Client:  cur\_id = 0  def \_\_init\_\_(self):  self.id = Client.cur\_id  Client.cur\_id += 1  class Generator:  def \_\_init\_\_(self, distribution):  self.distrib = distribution  self.time\_left = 0  def update(self):  self.time\_left -= dt  if self.time\_left <= EPS:  self.time\_left = self.distrib.generate()  return Client()  return None  class Manager:  def \_\_init\_\_(self, queue, distribution):  self.distrib = distribution  self.busy = False  self.queue = queue  self.current\_client = None  self.time\_left = 0  def accept\_client(self, client):  self.busy = True  self.current\_client = client  self.time\_left = self.distrib.generate()  def serve\_client(self):  self.queue.append(self.current\_client)  self.busy = False  self.current\_client = None  def update(self):  if self.busy:  self.time\_left -= dt  if self.time\_left <= EPS:  self.serve\_client()  class Mechanic:  def \_\_init\_\_(self, client\_queue, distribution):  self.distrib = distribution  self.busy = False  self.client\_queue = client\_queue  self.current\_client = None  self.time\_left = 0  def update(self):  res = ''  if self.busy:  self.time\_left -= dt  if self.time\_left <= EPS:  self.busy = False  self.current\_client = None  res = 'served'  if not self.busy and len(self.client\_queue) > 0:  self.current\_client = self.client\_queue.pop(0)  self.time\_left = self.distrib.generate()  self.busy = True  return res  def choose\_manager(managers):  for i in range(len(managers)):  if not managers[i].busy:  return i  return -1  def time\_step(generator, managers, mechanics, clients\_info, gen\_flag = True):  if gen\_flag:  client = generator.update()  if client:  clients\_info['visited'] += 1  man\_ind = choose\_manager(managers)  if man\_ind == -1:  clients\_info['denied'] += 1  else:  managers[man\_ind].accept\_client(client)  for man in managers:  man.update()  for mech in mechanics:  res = mech.update()  if res == 'served':  clients\_info['served'] += 1  def model(generator, managers, mechanics, total\_clients):  clients\_info = {'visited': 0, 'denied': 0, 'served': 0}  time = 0  while clients\_info['visited'] < total\_clients:  time\_step(generator, managers, mechanics, clients\_info)  time += dt  while clients\_info['denied'] + clients\_info['served'] < total\_clients:  time\_step(generator, managers, mechanics, clients\_info, False)  time += dt  return clients\_info, time  def main():  queue1 = []  queue2 = []    clients = Generator(EvenDistribution(14, 20))  managers = [  Manager(queue1, EvenDistribution(26, 34)),  Manager(queue2, EvenDistribution(28, 38))  ]  mechanics = [  Mechanic(queue1, EvenDistribution(10, 18)),  Mechanic(queue1, EvenDistribution(10, 20)),  Mechanic(queue1, EvenDistribution(15, 19)),  Mechanic(queue2, EvenDistribution(16, 20)),  Mechanic(queue2, EvenDistribution(18, 24))  ]  total\_clients = 1000  res, time = model(clients, managers, mechanics, total\_clients)  print('Клиентов посетило:', res['visited'])  print('Клиентов обслужено:', res['served'])  print('Клиентов отклонено:', res['denied'])  print('Модельное время:', round(time, 2))  print('Вероятность отказа:', round(res['denied'] / total\_clients, 3))  if \_\_name\_\_ == '\_\_main\_\_':  main() |

# Результаты работы

Результаты моделирования процесса обслуживания 1000 клиентов автосервиса представлены на рисунках 2-5. В приведенных примерах работы интервал времени поступления клиентов в автосервис равен 17 ± 3, 13 ± 3, 11±3 и 8 ± 2 минуты соответственно.

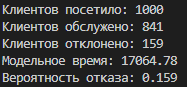


Рисунок 2 – Пример работы №1

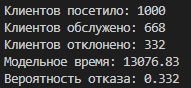


Рисунок 3 – Пример работы №2

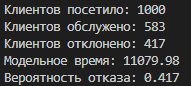


Рисунок 4 – Пример работы №3

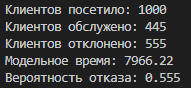


Рисунок 5 – Пример работы №4

По полученным результатам можно сделать вывод о том, что при уменьшении интервала времени поступления клиентов вероятность отказа в системе стремительно увеличивается – автосервис перестает справляться с нагрузкой.