Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Кубанский государственный университет»

Кафедра вычислительных технологий

**ОТЧЕТ**

о выполнении лабораторной работы №8

по дисциплине “Основы компьютерного моделирования”

Выполнил: ст. гр. 39/2

Петренко Д.А.

Проверил:

Патыковская М.В.

Вариант 5

Постановка задачи:

На регулировочный участок цеха через случайные интервалы времени поступают по два агрегата в среднем через каждые 30 мин. Первичная регулировка осуществляется для двух агрегатов одновременно и занимает около 30 мин. Если в момент прихода агрегатов предыдущая партия не была обработана, поступившие агрегаты на регулировку не принимаются. Агрегаты после первичной регулировки, получившие отказ, поступают в промежуточный накопитель. Из накопителя агрегаты, прошедшие первичную регулировку, поступают попарно на вторичную регулировку, которая выполняется в среднем за 30 мин, а не прошедшие первичную регулировку поступают на полную, которая занимает 100 мин для одного агрегата. Все величины, заданные средними значениями, распределены по экспоненциальному закону;

Смоделировать работу участка в течение 100 ч. Определить:

1) вероятность отказа в первичной регулировке;

2) загрузку накопителя агрегатами, нуждающихся в полной регулировке;

3) ввести в систему накопитель, обеспечивающий безотказное обслуживание поступающих агрегатов и определить его параметры.

Математическая модель: Данная система может быть моделирована с помощью системы массового обслуживания (СМО) с несколькими узлами обработки и ожидания. В данном случае, у нас есть три узла: предварительная обработка, сборка и регулировка,

Основные характеристики СМО:

Среднее время поступления двух агрегатов на регулировочный участок цеха: t\_1 - распределено экспоненциально со средним значением 30 минут.

Среднее время первичной регулировки: t\_2 распределено экспоненциально со средним значением 30 минут.

Среднее время вторичной регулировки: t\_3 - распределено экспоненциально со средним значением 30 минут.

Среднее время полной вторичной регулировки: t\_4 - распределено экспоненциально со средним значением 100 минут.

Программа для заданных задачей входных данных:

import numpy as np

import pandas as pd

# 100 часов

sim\_time = 100 \* 60

#время поступления

t\_1 = 30

#Первичная

t\_2 = 30

#Вторичная, если прошли первичную

t\_3 = 30

# Полная вторичная

t\_4 = 100

# Переменные и счётчики

time = 0

status = 0

failure\_count = 0

full\_reg\_count = 0

intermediate\_storage = []

# Функция для генерации случайных чисел из экспоненциального распределения

def exp\_regulation(mean\_time):

return np.random.exponential(mean\_time)

# Функция имитации процесса регулирования

def simulate\_regulation():

global time, status, failure\_count, full\_reg\_count, intermediate\_storage

while time < sim\_time:

# Генерация случайного времени прибытия для двух агрегатов

time\_to\_next\_arrival = exp\_regulation(t\_1)

# Проверка, регулировались ли предыдущие агрегаты

if status == 0:

# Выполнение первичной регулировки

time\_to\_primary\_regulation = exp\_regulation(t\_2)

time += time\_to\_primary\_regulation

# Проверка, не сработала ли первичная регулировка

if np.random.random() < 0.05:

# Полная вторичная регулировка

failure\_count += 1

time\_to\_secondary\_regulation = exp\_regulation(t\_4)

time += time\_to\_secondary\_regulation

status = 1

else:

# вторичная регулировка

time\_to\_secondary\_regulation = exp\_regulation(t\_3)

time += time\_to\_secondary\_regulation

status = 1

else:

# Добавить агрегаты в промежуточное хранилище

intermediate\_storage.append(2)

status = 0

# Проверка промежуточного хранилища на заполненность, требующую полного регулирования

if len(intermediate\_storage) >= 2:

full\_reg\_count += len(intermediate\_storage) // 2

intermediate\_storage = intermediate\_storage[len(intermediate\_storage) % 2:]

time += time\_to\_next\_arrival

return failure\_count, full\_reg\_count

# Старт симуляции

failure\_count, full\_reg\_count = simulate\_regulation()

# Рассчёт вероятности отказа первичной регуляции

failure\_prob = failure\_count / (sim\_time / 60 \* 2 / 30)

# Рассчёт среднего количество заполнителей в промежуточном хранилище, требующих полного регулирования

avg\_full\_reg = full\_reg\_count / (sim\_time / 60 \* 2 / 30)

# Рассчёт параметров системы промежуточного хранения для обеспечения безотказной работы агрегатов

max\_arrivals = (1 - failure\_prob) \* 2 / 3

avg\_arrivals = avg\_full\_reg / (sim\_time / 60)

# Вывод

print("Результат:")

print(f"Вероятность отказа первичной регулировке: {failure\_prob:.2f}")

print(f"Среднее количество агрегатов на промежуточном хранении, требующих полного регулирования: {avg\_full\_reg:.2f}")

print(f"Максимальное количество поступлений в систему промежуточного хранения для обеспечения отсутствия выхода из строя агрегатов: {max\_arrivals:.2f}")

print(f"Среднее количество поступлений в систему промежуточного хранения для обеспечения отсутствия отказов агрегатов: {avg\_arrivals:.2f}")

Результаты:

Вероятность отказа первичной регулировке: 0.75

Среднее количество агрегатов на промежуточном хранении, требующих полного регулирования: 14.70

Максимальное количество поступлений в систему промежуточного хранения для обеспечения отсутствия выхода из строя агрегатов: 0.17

Среднее количество поступлений в систему промежуточного хранения для обеспечения отсутствия отказов агрегатов: 0.15

После выполнения моделирования работы участка программа выведет результат в консоль. Рассмотрим возможные выводы и их анализ:

Вероятность отказа первичной регулировке:

Среднее количество агрегатов на промежуточном хранении, требующих полного регулирования.

Максимальное количество поступлений в систему промежуточного хранения для обеспечения отсутствия выхода из строя агрегатов: Этот вывод указывает, что .

Среднее количество поступлений в систему промежуточного хранения для обеспечения отсутствия отказов агрегатов: Этот вывод указывает.