

# **ЛАБОРАТОРНАЯ РАБОТА №1.**

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ-КАЛЬКУЛЯТОРА**

### **1. НАЗНАЧЕНИЕ И КРАТКАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

В процессе выполнения данной работы студенты применяют полученные знания ООП и отрабатывают навыки разработки вычислительных алгоритмов на языке JAVA.

### **2. ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ**

В ходе лабораторной работы необходимо разработать приложение-калькулятор, способное работать, как с арабскими, так и с римскими числами. Калькулятор должен осуществлять следующие арифметические операции:

- Сложение (+);
- Вычитание (-);
- Умножение (\*);
- Деление (/).

### **3. ЗАДАНИЕ НА РАБОТУ В ЛАБОРАТОРИИ**

В рамках лабораторной работы необходимо:

1. Создать класс “Main” с единственным методом “main” со следующим функционалом:
  - Считывание из консоли арифметического выражения;
  - Определение системы счисления: арабские или римские цифры;
  - В зависимости от системы счисления создание подходящего объекта для проведения расчётов;
  - Вывод в консоль результата расчётов.

2. Разработать класс "NumberIdentifier", обладающий двумя статическими методами:
  - boolean isArabicNumber(String number) - возвращает true, если число представлено в арабской системе счисления и false, если в римской;
  - boolean isRomanNumber(String number) - возвращает true, если число представлено в римской системе счисления и false, если в арабской.
3. Разработать абстрактный класс "Calculations", который содержит следующие методы:
  - addition() – метод, выполняющий сложение;
  - subtraction() – метод, выполняющий вычитание;
  - multiplication() – метод, выполняющий умножение;
  - division() – метод выполняющий деление.
4. Разработать два класса наследника класса "Calculations" для работы с арабскими числами "ArabicCalcs" и римскими числами "RomanCalcs".

## 4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ

К пункту 1. Перед началом работы создадим новый проект.

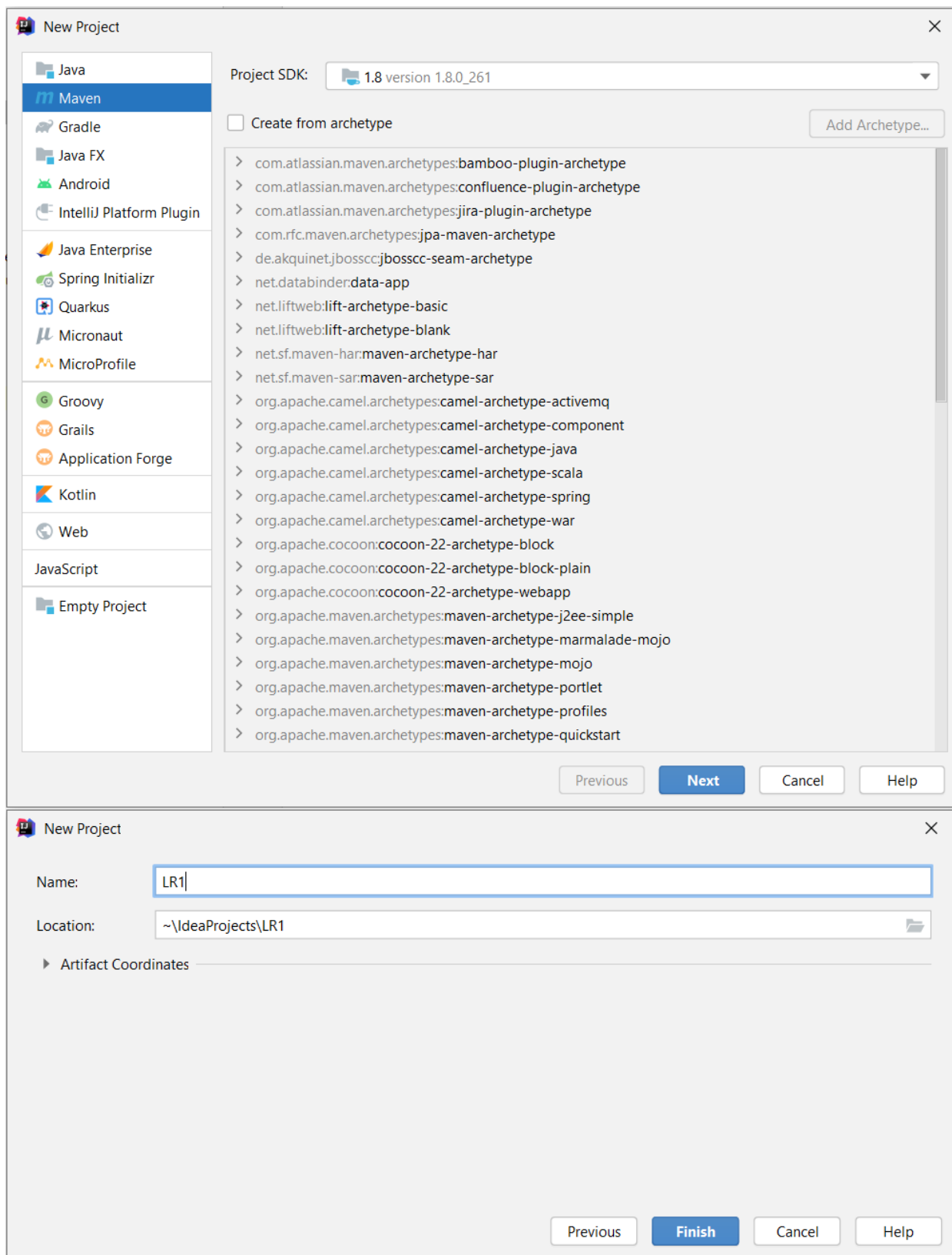


Рисунок 1 – Создание нового проекта на базе шаблона Maven

Далее в созданном проекте в папке `src.main.java` создаём новую папку, в которой будут храниться классы.

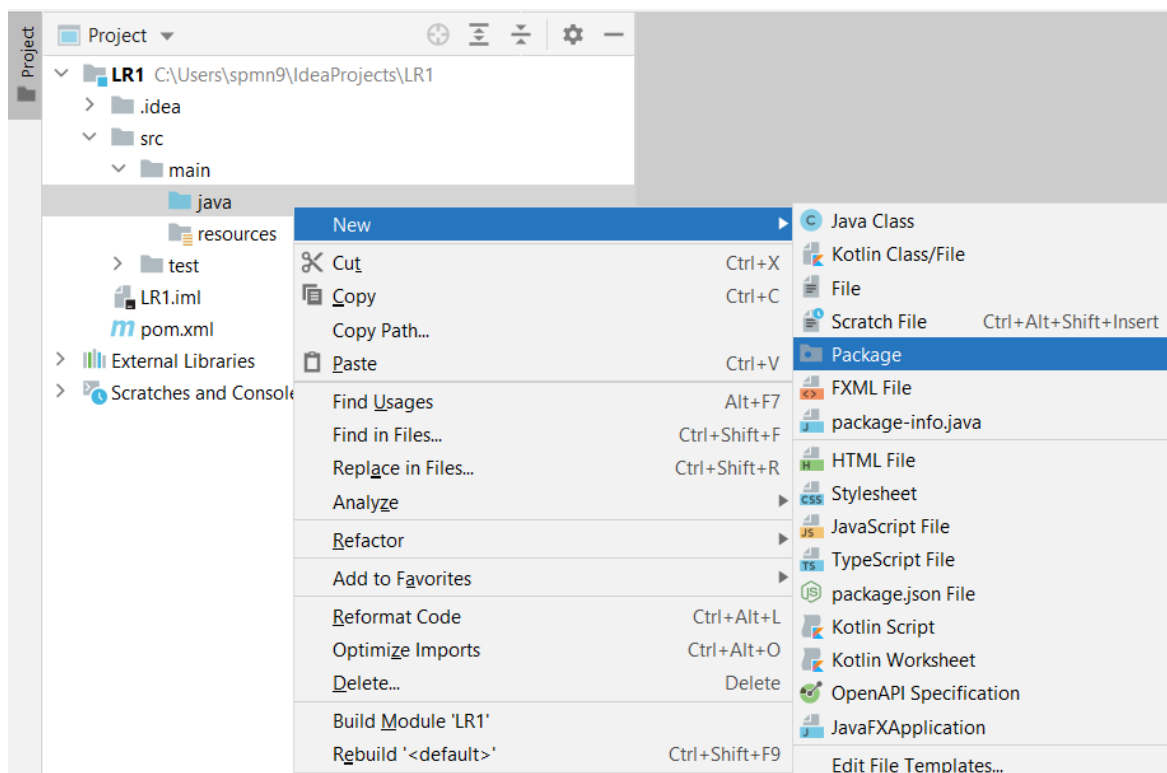


Рисунок 2 – Создание новой папки

Далее создаём новый класс “Main”:

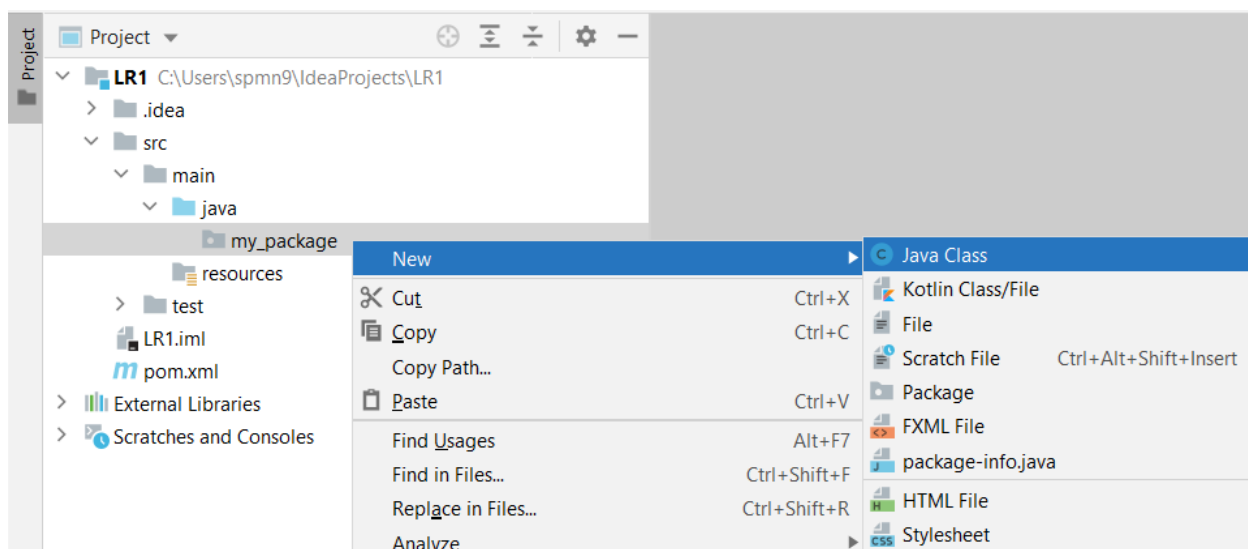


Рисунок 3 – Создание нового класса

Для того, чтобы не писать сигнатуру метода “main” вручную достаточно ввести “psvm” и нажать “Tab”.

Считывание арифметического выражения из консоли необходимо осуществлять с помощью класса “Scanner”.

Для определения системы счисления необходимо будет воспользоваться статическими методами из класса “NumberIdentifier”, описанного в пункте 2.

Подходящим объектом для вычисления является объект класса “ArabicCalcs”, если были введены арабские числа, “RomanCalcs”, если были введены римские числа.

Гарантируется, что на вход будут поданы два числа не более 10, как арабские, так и римские. Также гарантируется, что в выражении будет одна из арифметических операций, описанных выше. Результатом вычисления может быть любое число

Если на вход поданы числа более 10, либо из разных систем счисления, например,  $5 * V$ , то тогда программа должна остановить своё выполнение и вывести в консоль сообщение “Неверный формат чисел”.

Вывод результата в консоль необходимо осуществить в той же системе счисления, что и входные данные, так например, если на вход были поданы арабские числа, то результат необходимо вывести в арабских числах, если входные данные в римских числах, следовательно, в результате необходимо вывести римское число.

К пункту 2. Для определения является ли число римским можно воспользоваться конструкцией “switch case”, в которой рассмотреть все возможные варианты чисел, которые могут быть поданы на вход (от 1 до 10).

К пункту 4. Помимо переопределённых методов из класса “Calculations” при необходимости можно создать дополнительные методы для внутренних действий.

Метод “subtraction()” при работе с арабскими числами может в качестве результата вернуть отрицательное число, при работе с римскими числами

необходимо вывести в консоль “Неположительный результат”, если разница чисел  $\leq 0$ .

Метод “division()” во всех случаях должен возвращать результат деления с округлением вниз.

Поскольку результат необходимо выводить в той же системе счисления, что и вводные данные, то необходимо будет реализовать алгоритм перевода арабских чисел в римские. Поскольку существует ограничение на размер входных чисел, алгоритм для перевода достаточно осуществить до числа 100, поскольку 100 – максимальное число, которое можно получить в ходе работы с таким калькулятором.

Примеры работы программы:

Ввод: 5\*6

Вывод: 30

Ввод: 9/4

Вывод: 2

Ввод: 5+V

Вывод: “Неверный формат чисел”

Ввод: VI\*III

Вывод: XVIII

Ввод: III-X

Вывод: “Неположительный результат”

## **ЛАБОРАТОРНАЯ РАБОТА №2.**

### **Кооперация агентов для решения задачи оптимизации**

#### **1. НАЗНАЧЕНИЕ И КРАТКАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

В процессе выполнения данной работы студенты получают базовые навыки создания мультиагентной системы на основе кооперации агентов, т.е. решение общей задачи.

#### **2. ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ**

Понятие кооперации агентов играет центральную роль в МАС. Кооперация агентов – это форма взаимодействия агентов, при которой все агенты преследуют общую цель и «жертвуют» своими интересами для достижения оптимального состояния в целом. При кооперации агенты распределяют между собой выполняемые функции, определяют роли для каждого из агентов, выполняют свои обязанности в рамках общей целевой функции. В общем случае понятие «координация» можно разложить на следующие составляющие:

- Сотрудничество агентов
- Координация действий
- Разрешение конфликтов

В данной работе будет рассмотрена система из группы агентов, каждый из которых имеет свою целевую функцию, заданную простейшим уравнением (линейным, квадратичным, синусоидальным и т.д.). Задача агентов в процессе кооперации определить такое состояние системы (значение аргумента функции ( $X$ ), чтобы найти экстремум суммарной функции.

Данная постановка задачи является упрощением задачи распределенной оптимизации потерь электроэнергии в группе энергокласетров (группа подстанций и станций), в состав которых входят средства компенсации реактивной мощности (СКРМ) и средства регулирования напряжения (СРН).

Данная система была разработана в рамках проекта ПТК РИСУ НРМ (Центр НТИ «Технологии транспортировки электроэнергии и распределенных интеллектуальных энергосистем»). В системе каждый агент представляет энергокластер и имеет ряд инструментов (СКРМ и СРН) для изменения режимных параметров. Агенты проводят внутреннюю оптимизацию режима (т.е. решают оптимизационную задачу для своего участка) и способны инициировать расчет под заданные критерии (электрические параметры на граничных линиях). Обмениваясь информацией по граничным линиям, агенты стремятся определить такое состояние всей системы в целом, чтобы суммарные потери активной мощности были минимальны.

В лабораторной работе процесс оптимизации потерь активной мощности у агента (расчет целевой функции для данного энергокластера) заменен на вычисление уравнения, заданного у агента. Поиск оптимального состояния системы – экстремум суммарной функции является примером того, как агенты взаимодействуя между собой способны распределено находить решение задачи.

Плюсами такого способа решения задачи в сравнении с традиционным централизованным подходом будут:

- Простое горизонтальное расширение системы: добавление нового узла не требует изменения алгоритма, агенты сами определяют новый узел и будут его учитывать в процессе оптимизации
- Самодиагностика и динамическая реконфигурация: при удалении агента система также автоматически определяет изменившиеся условия и перестраивает механизм оптимизации без ручного изменения алгоритма
- Возможность учета интереса каждого агента в зависимости от заданной целевой функции



### **3. ЗАДАНИЕ НА РАБОТУ В ЛАБОРАТОРИИ**

В рамках лабораторной работы необходимо:

1. разработать класс агента `FunctionAgent.class`;
2. реализовать следующие поведения агента:
  - a. принятие запроса от другого агентов на расчет значения функции, заданной агенту;
  - b. отправка запросов на расчет значений функций других агентов, принятия ответов и оценки полученных результатов;
  - c. передача очереди инициирования проведения расчетов;
  - d. принятие сообщения о передачи очереди для проведения расчетов .
3. реализовать вспомогательные классы для каждого из агентов для расчёта целевой функции;
4. Запустить работу системы с тремя агентами в соответствии с вариантом. Вывести найденный результат (значение функции и аргумента) в консоль.

#### 4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ

К пункту 2. Одним из возможных подходов к решению задачи является следующий алгоритм:

1. Первый агент производит выбор начальной точки ( $X$ ) случайным образом в некотором диапазоне, а также задает первоначальное значение параметра  $\delta$  – сдвиг, который служит для определения возрастания суммарного значения функций всех агентов. Первый агент становится агентом-инициатором.
2. Агент инициатор отправляет сообщение всем известным агентам для выполнения расчета графика функций каждого из агентов для трех точек:
  - $X - \delta$
  - $X$
  - $X + \delta$
3. Агент-инициатор рассчитывает значение своей функции для этих же точек.
4. Агент-инициатор определяет экстремум суммарного значения функций на основе полученных ответов и своих расчетов и фиксирует полученное для нее значение аргумента:
  - Если новой точкой является  $X - \delta$  или  $X + \delta$ , то агент инициатор передает сообщение следующему агенту, выбранному случайным образом с информацией о новой точкой и текущей  $\delta$
  - Если новой точкой остался  $X$ , то необходимо уменьшить величину  $\delta$  в два раза
5. Шаги 2-4 повторяются до тех пор, пока значение  $\delta$  не снизится до заданной точности (например, 0.01)

К пункту 3. Целесообразно создать 3 статических метода в вспомогательном классе, принимающие на вход значение аргумента, и возвращающие значение заданной функции от этого аргумента.

## 5. ВАРИАНТЫ ЗАДАНИЙ

Таблица 1: варианты заданий

№ Варианта	Агент 1	Агент 2	Агент 3	Экстремум
1	$y = -x^2 + 5$	$y = 2x + 2$	$y = \sin(x)$	max
2	$y = e^{-0.5x}$	$y = 0.5x + w$	$y = \cos(x)$	min
3	$y = e^{0.3x}$	$y = x^2$	$y = \sin(x)$	min
4	$y = e^{0.2x}$	$y = 2^{-1x}$	$y = \cos(x)$	min
5	$y = -0.5x^2 - 4$	$y = 2^{-0.1x}$	$y = \cos(x)$	max

# **ЛАБОРАТОРНАЯ РАБОТА №3**

## **Поиск лучшего пути**

### **1. НАЗНАЧЕНИЕ И КРАТКАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

Выполнение лабораторной работы предполагает первоначальное представление о теории графов, ее определениях и некоторых алгоритмах решения проблем поиска пути с помощью этой теории.

Целью работы является составление и написание алгоритма поиска кратчайшего пути в графе на языке Java с использованием технологии агентного взаимодействия. Каждая вершина графа – самостоятельный агент. Каждое ребро графа представляет собой связь агентов и имеет уникальный вес. Агент-вершина обладает информацией о связях только с соседними агентами.

### **2. ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ**

Теория графов - один из обширнейших разделов дискретной математики, широко применяющийся в программировании и изучении электрических цепей, коммуникации. Теория графов систематически и последовательно изучает свойства графов, о которых можно сказать, что они состоят из множеств точек и множеств линий, отображающих связи между этими точками. Основателем теории графов считается Леонард Эйлер (1707-1882), решивший в 1736 году известную в то время задачу о кёнигсбергских мостах. Пример графа приведен ниже (Рисунок )

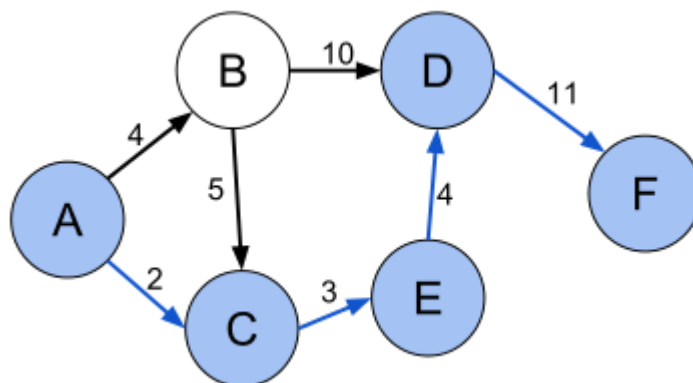


Рисунок 4 - Взвешенный неориентированный граф

Задача о кратчайшем пути — задача поиска самого короткого пути (цепи) между двумя точками (вершинами) на графе, в которой минимизируется сумма весов рёбер, составляющих путь. Задача о кратчайшем пути является одной из важнейших классических задач теории графов.

В данной лабораторной работе будет необходимо разработать распределенную систему поиска кратчайшего пути в графе, вершинами которого являются агенты, а ребрами – заданные веса между ними.

Пример расположения агентов и связей между ними представлены на рисунке ниже (Рисунок )

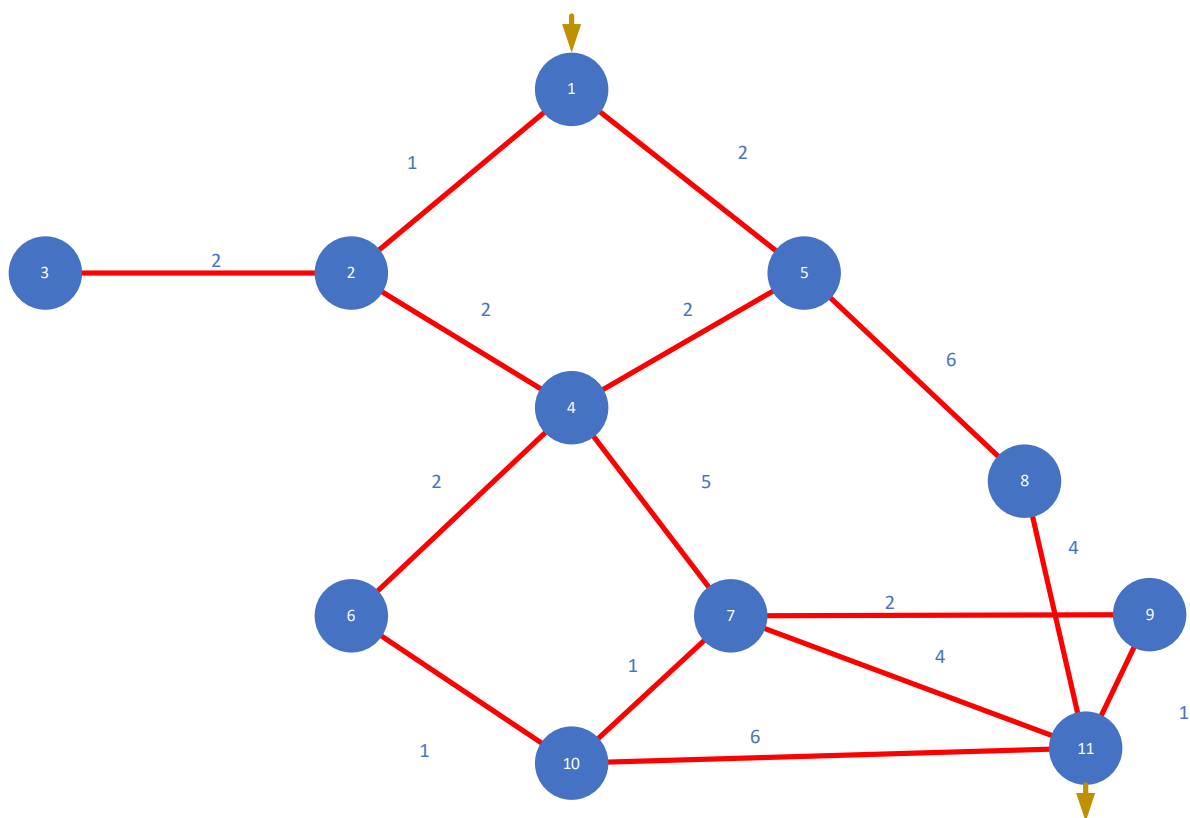


Рисунок 5 - пример задания для поиска кратчайшего пути

Начальными данными для решения являются:

- Начальный агент
- Конечный агент
- Список соседей и вес связи для каждого из соседей для каждого агента

Главным ограничением задачи распределенного поиска является наличие информации у агента только о своих соседях и о весе связи с каждым из них т.е. сбор и передача информации в одном из узлов, а затем ее последующая обработка для нахождения кратчайшего пути запрещена. Поэтому поиск для решения задачи будет необходимо произвести с помощью коммуникации агентов.

### **3. ЗАДАНИЕ НА РАБОТУ В ЛАБОРАТОРИИ**

В рамках лабораторной работы необходимо:

1. Разработать класс агента NodeAgent.class
2. Разработать конфигурационный файл агента NodeAgent
3. Реализовать необходимые поведения для поиска заданного узла:
  - a. Реализовать поведение инициации процесса поиска
  - b. Реализовать поведение обработки запроса поиска
  - c. Реализовать поведение обратной передачи сообщения при нахождении заданного агента
4. Провести опыты поиска с различными начальными и искомыми агентами

### **4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

#### К пункту 1.

Каждый узел должен быть представлен экземпляром класса NodeAgent, в том числе начальный и конечные агенты. При начале работы агент должен считать свой конфигурационный файл и при наличии у него метки о начале работы произвести старт поиска заданного искомого агента

#### К пункту 2.

Конфигурационный файл агента должен содержать информацию о том, является ли агент инициатором и информацию об искомом агенте, если он им действительно является. Также в конфигурационном файле содержится список соседних агентов и вес связей между ними в соответствии с вариантом задания.

#### К пункту 3.

Поведение, иницирующее процесс поиска агента, должно выполняться при наличии соответствующего поля в конфигурационном файле агента. В рамках этого поведения агент отправляет первые сообщения своим соседям и ожидает некоторое время, за которое агенты отправляют ему информацию о проведенном поиске. Передаваемые сообщения должны хранить информацию

о цепочке отправителей, а также идентификатор искомого агента. После истечения времени агент анализирует результат поиска, определяет кратчайший путь и выписывает в консоль результат.

Поведение обработки запроса поиска представляет собой механизм приема сообщения от любого агента и анализ своих соседей для того, чтобы передать им сообщение поиска. Анализ соседей заключается в поиске тех агентов, кому необходимо отправить сообщение, и тех, которых необходимо исключить для того, чтобы избежать заикливания алгоритма. Если при оценке соседей не было найдено ни одного агента для отправки, данный путь считается тупиковым и агенту-инициатору отправляется путь с отметкой о неудаче. Во время каждой передачи сообщения необходимо записать информацию об агенте, через которого происходит передача с указанием веса. Эта информация будет использована для возможности обратной передачи сообщений, а также для подсчета итогового величины пути.

Поведение обратной отправки сообщений необходимо для получения агентом-инициатором результатов поиска от других агентов. После нахождения искомого агента или при неудачном поиске для какого-то конкретного пути, необходимо сформировать сообщение с информацией о рассматриваемой цепочке агентов и о весе, который был определен в процессе поиска.

#### К пункту 4.

Провести проверку разработанного алгоритма с несколькими разными агентами-инициаторами и искомыми агентами



## 5. ВАРИАНТЫ ЗАДАНИЙ

В качестве исходных данных каждому студенту даётся матрица смежности взвешенного графа.

Таблица 2: варианты заданий

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
1	1	0	4	5	2	10	0	0	0	0	0	0	0
	2	4	0	22	2	19	0	0	0	0	0	0	0
	3	5	22	0	22	23	0	0	0	0	0	0	0
	4	2	2	22	0	17	13	12	25	0	0	0	0
	5	10	19	23	17	0	25	19	17	0	0	0	0
	6	0	0	0	13	25	0	20	9	22	0	0	0
	7	0	0	0	12	19	20	0	1	19	0	0	0
	8	0	0	0	25	17	9	1	0	21	0	0	0
	9	0	0	0	0	0	22	19	21	0	18	2	23
	10	0	0	0	0	0	0	0	0	18	0	26	25
	11	0	0	0	0	0	0	0	0	2	26	0	15
	12	0	0	0	0	0	0	0	0	23	25	15	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
2	1	0	11	21	19	21	0	0	0	0	0	0	0
	2	11	0	14	5	12	0	0	0	0	0	0	0
	3	21	14	0	26	7	0	0	0	0	0	0	0
	4	19	5	26	0	19	4	22	26	0	0	0	0
	5	21	12	7	19	0	10	5	6	0	0	0	0
	6	0	0	0	4	10	0	11	27	17	0	0	0
	7	0	0	0	22	5	11	0	23	1	0	0	0
	8	0	0	0	26	6	27	23	0	12	0	0	0
	9	0	0	0	0	0	17	1	12	0	4	1	11
	10	0	0	0	0	0	0	0	0	4	0	20	9
	11	0	0	0	0	0	0	0	0	1	20	0	17
	12	0	0	0	0	0	0	0	0	11	9	17	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
3	1	0	5	2	6	22	0	0	0	0	0	0	0
	2	5	0	21	22	24	0	0	0	0	0	0	0
	3	2	21	0	12	14	0	0	0	0	0	0	0
	4	6	22	12	0	14	15	26	3	0	0	0	0
	5	22	24	14	14	0	21	19	14	0	0	0	0
	6	0	0	0	15	21	0	20	6	18	0	0	0
	7	0	0	0	26	19	20	0	4	11	0	0	0
	8	0	0	0	3	14	6	4	0	13	0	0	0
	9	0	0	0	0	0	18	11	13	0	8	18	5
	10	0	0	0	0	0	0	0	0	8	0	17	8
	11	0	0	0	0	0	0	0	0	18	17	0	25
	12	0	0	0	0	0	0	0	0	5	8	25	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
4	1	0	7	24	14	16	0	0	0	0	0	0	0
	2	7	0	17	20	25	0	0	0	0	0	0	0
	3	24	17	0	15	2	0	0	0	0	0	0	0
	4	14	20	15	0	11	23	11	17	0	0	0	0
	5	16	25	2	11	0	23	15	6	0	0	0	0
	6	0	0	0	23	23	0	14	6	20	0	0	0
	7	0	0	0	11	15	14	0	14	9	0	0	0
	8	0	0	0	17	6	6	14	0	17	0	0	0
	9	0	0	0	0	0	20	9	17	0	25	6	25
	10	0	0	0	0	0	0	0	0	25	0	6	1
	11	0	0	0	0	0	0	0	0	6	6	0	20
	12	0	0	0	0	0	0	0	0	25	1	20	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
5	1	0	18	6	8	4	0	0	0	0	0	0	0
	2	18	0	23	8	25	0	0	0	0	0	0	0
	3	6	23	0	14	15	0	0	0	0	0	0	0
	4	8	8	14	0	26	13	18	5	0	0	0	0
	5	4	25	15	26	0	7	12	16	0	0	0	0
	6	0	0	0	13	7	0	19	5	7	0	0	0
	7	0	0	0	18	12	19	0	25	7	0	0	0
	8	0	0	0	5	16	5	25	0	7	0	0	0
	9	0	0	0	0	0	7	7	7	0	10	16	9
	10	0	0	0	0	0	0	0	0	10	0	18	17
	11	0	0	0	0	0	0	0	0	16	18	0	10
	12	0	0	0	0	0	0	0	0	9	17	10	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
6	1	0	16	2	8	25	0	0	0	0	0	0	0
	2	16	0	13	15	4	0	0	0	0	0	0	0
	3	2	13	0	15	24	0	0	0	0	0	0	0
	4	8	15	15	0	17	14	23	19	0	0	0	0
	5	25	4	24	17	0	1	15	21	0	0	0	0
	6	0	0	0	14	1	0	3	8	23	0	0	0
	7	0	0	0	23	15	3	0	8	16	0	0	0
	8	0	0	0	19	21	8	8	0	14	0	0	0
	9	0	0	0	0	0	23	16	14	0	16	22	4
	10	0	0	0	0	0	0	0	0	16	0	19	12
	11	0	0	0	0	0	0	0	0	22	19	0	5
	12	0	0	0	0	0	0	0	0	4	12	5	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
7	1	0	14	19	11	19	0	0	0	0	0	0	0
	2	14	0	16	5	19	0	0	0	0	0	0	0
	3	19	16	0	26	13	0	0	0	0	0	0	0
	4	11	5	26	0	24	4	18	21	0	0	0	0
	5	19	19	13	24	0	19	24	26	0	0	0	0
	6	0	0	0	4	19	0	4	17	18	0	0	0
	7	0	0	0	18	24	4	0	24	16	0	0	0
	8	0	0	0	21	26	17	24	0	5	0	0	0
	9	0	0	0	0	0	18	16	5	0	1	18	11
	10	0	0	0	0	0	0	0	0	1	0	7	18
	11	0	0	0	0	0	0	0	0	18	7	0	13
	12	0	0	0	0	0	0	0	0	11	18	13	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
8	1	0	12	22	1	2	0	0	0	0	0	0	0
	2	12	0	20	24	17	0	0	0	0	0	0	0
	3	22	20	0	3	14	0	0	0	0	0	0	0
	4	1	24	3	0	16	23	1	25	0	0	0	0
	5	2	17	14	16	0	22	24	25	0	0	0	0
	6	0	0	0	23	22	0	1	12	23	0	0	0
	7	0	0	0	1	24	1	0	12	1	0	0	0
	8	0	0	0	25	25	12	12	0	18	0	0	0
	9	0	0	0	0	0	23	1	18	0	24	17	2
	10	0	0	0	0	0	0	0	0	24	0	10	11
	11	0	0	0	0	0	0	0	0	17	10	0	17
	12	0	0	0	0	0	0	0	0	2	11	17	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
9	1	0	13	8	2	21	0	0	0	0	0	0	0
	2	13	0	1	16	14	0	0	0	0	0	0	0
	3	8	1	0	4	1	0	0	0	0	0	0	0
	4	2	16	4	0	7	12	3	24	0	0	0	0
	5	21	14	1	7	0	24	24	19	0	0	0	0
	6	0	0	0	12	24	0	23	3	6	0	0	0
	7	0	0	0	3	24	23	0	13	16	0	0	0
	8	0	0	0	24	19	3	13	0	19	0	0	0
	9	0	0	0	0	0	6	16	19	0	8	20	21
	10	0	0	0	0	0	0	0	0	8	0	12	4
	11	0	0	0	0	0	0	0	0	20	12	0	3
	12	0	0	0	0	0	0	0	0	21	4	3	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
10	1	0	18	10	21	27	0	0	0	0	0	0	0
	2	18	0	12	7	3	0	0	0	0	0	0	0
	3	10	12	0	23	2	0	0	0	0	0	0	0
	4	21	7	23	0	8	15	11	1	0	0	0	0
	5	27	3	2	8	0	4	14	20	0	0	0	0
	6	0	0	0	15	4	0	26	5	12	0	0	0
	7	0	0	0	11	14	26	0	18	22	0	0	0
	8	0	0	0	1	20	5	18	0	16	0	0	0
	9	0	0	0	0	0	12	22	16	0	10	23	11
	10	0	0	0	0	0	0	0	0	10	0	3	22
	11	0	0	0	0	0	0	0	0	23	3	0	24
	12	0	0	0	0	0	0	0	0	11	22	24	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
11	1	0	15	25	15	1	0	0	0	0	0	0	0
	2	15	0	24	17	6	0	0	0	0	0	0	0
	3	25	24	0	25	10	0	0	0	0	0	0	0
	4	15	17	25	0	12	20	24	18	0	0	0	0
	5	1	6	10	12	0	16	19	5	0	0	0	0
	6	0	0	0	20	16	0	13	8	10	0	0	0
	7	0	0	0	24	19	13	0	26	15	0	0	0
	8	0	0	0	18	5	8	26	0	4	0	0	0
	9	0	0	0	0	0	10	15	4	0	23	26	1
	10	0	0	0	0	0	0	0	0	23	0	2	8
	11	0	0	0	0	0	0	0	0	26	2	0	14
	12	0	0	0	0	0	0	0	0	1	8	14	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
12	1	0	24	2	18	24	0	0	0	0	0	0	0
	2	24	0	21	27	16	0	0	0	0	0	0	0
	3	2	21	0	19	20	0	0	0	0	0	0	0
	4	18	27	19	0	26	10	1	8	0	0	0	0
	5	24	16	20	26	0	7	9	13	0	0	0	0
	6	0	0	0	10	7	0	16	10	5	0	0	0
	7	0	0	0	1	9	16	0	18	8	0	0	0
	8	0	0	0	8	13	10	18	0	24	0	0	0
	9	0	0	0	0	0	5	8	24	0	5	19	5
	10	0	0	0	0	0	0	0	0	5	0	6	8
	11	0	0	0	0	0	0	0	0	19	6	0	12
	12	0	0	0	0	0	0	0	0	5	8	12	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
13	1	0	6	1	12	1	0	0	0	0	0	0	0
	2	6	0	8	12	17	0	0	0	0	0	0	0
	3	1	8	0	18	27	0	0	0	0	0	0	0
	4	12	12	18	0	13	18	4	26	0	0	0	0
	5	1	17	27	13	0	23	13	1	0	0	0	0
	6	0	0	0	18	23	0	6	3	17	0	0	0
	7	0	0	0	4	13	6	0	11	4	0	0	0
	8	0	0	0	26	1	3	11	0	20	0	0	0
	9	0	0	0	0	0	17	4	20	0	2	20	1
	10	0	0	0	0	0	0	0	0	2	0	1	11
	11	0	0	0	0	0	0	0	0	20	1	0	13
	12	0	0	0	0	0	0	0	0	1	11	13	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
14	1	0	15	27	21	25	0	0	0	0	0	0	0
	2	15	0	14	13	11	0	0	0	0	0	0	0
	3	27	14	0	27	27	0	0	0	0	0	0	0
	4	21	13	27	0	27	6	5	23	0	0	0	0
	5	25	11	27	27	0	20	4	15	0	0	0	0
	6	0	0	0	6	20	0	25	24	16	0	0	0
	7	0	0	0	5	4	25	0	7	22	0	0	0
	8	0	0	0	23	15	24	7	0	6	0	0	0
	9	0	0	0	0	0	16	22	6	0	17	27	17
	10	0	0	0	0	0	0	0	0	17	0	2	20
	11	0	0	0	0	0	0	0	0	27	2	0	7
	12	0	0	0	0	0	0	0	0	17	20	7	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
15	1	0	4	20	6	15	0	0	0	0	0	0	0
	2	4	0	20	19	12	0	0	0	0	0	0	0
	3	20	20	0	4	10	0	0	0	0	0	0	0
	4	6	19	4	0	25	20	19	8	0	0	0	0
	5	15	12	10	25	0	3	10	21	0	0	0	0
	6	0	0	0	20	3	0	5	12	1	0	0	0
	7	0	0	0	19	10	5	0	17	19	0	0	0
	8	0	0	0	8	21	12	17	0	18	0	0	0
	9	0	0	0	0	0	1	19	18	0	22	19	2
	10	0	0	0	0	0	0	0	0	22	0	19	4
	11	0	0	0	0	0	0	0	0	19	19	0	16
	12	0	0	0	0	0	0	0	0	2	4	16	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
16	1	0	13	25	23	1	0	0	0	0	0	0	0
	2	13	0	20	20	10	0	0	0	0	0	0	0
	3	25	20	0	6	26	0	0	0	0	0	0	0
	4	23	20	6	0	11	19	20	19	0	0	0	0
	5	1	10	26	11	0	1	14	16	0	0	0	0
	6	0	0	0	19	1	0	12	23	12	0	0	0
	7	0	0	0	20	14	12	0	13	12	0	0	0
	8	0	0	0	19	16	23	13	0	18	0	0	0
	9	0	0	0	0	0	12	12	18	0	12	9	9
	10	0	0	0	0	0	0	0	0	12	0	20	12
	11	0	0	0	0	0	0	0	0	9	20	0	25
	12	0	0	0	0	0	0	0	0	9	12	25	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
17	1	0	18	9	22	12	0	0	0	0	0	0	0
	2	18	0	5	4	14	0	0	0	0	0	0	0
	3	9	5	0	11	1	0	0	0	0	0	0	0
	4	22	4	11	0	23	1	15	12	0	0	0	0
	5	12	14	1	23	0	6	5	14	0	0	0	0
	6	0	0	0	1	6	0	9	17	3	0	0	0
	7	0	0	0	15	5	9	0	13	9	0	0	0
	8	0	0	0	12	14	17	13	0	16	0	0	0
	9	0	0	0	0	0	3	9	16	0	24	12	2
	10	0	0	0	0	0	0	0	0	24	0	26	14
	11	0	0	0	0	0	0	0	0	12	26	0	2
	12	0	0	0	0	0	0	0	0	2	14	2	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
18	1	0	12	4	1	13	0	0	0	0	0	0	0
	2	12	0	11	27	2	0	0	0	0	0	0	0
	3	4	11	0	7	1	0	0	0	0	0	0	0
	4	1	27	7	0	4	20	9	13	0	0	0	0
	5	13	2	1	4	0	6	13	19	0	0	0	0
	6	0	0	0	20	6	0	13	8	6	0	0	0
	7	0	0	0	9	13	13	0	17	4	0	0	0
	8	0	0	0	13	19	8	17	0	3	0	0	0
	9	0	0	0	0	0	6	4	3	0	4	21	17
	10	0	0	0	0	0	0	0	0	4	0	2	23
	11	0	0	0	0	0	0	0	0	21	2	0	18
	12	0	0	0	0	0	0	0	0	17	23	18	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
19	1	0	15	26	22	25	0	0	0	0	0	0	0
	2	15	0	1	25	15	0	0	0	0	0	0	0
	3	26	1	0	13	18	0	0	0	0	0	0	0
	4	22	25	13	0	11	1	19	2	0	0	0	0
	5	25	15	18	11	0	22	5	11	0	0	0	0
	6	0	0	0	1	22	0	2	15	7	0	0	0
	7	0	0	0	19	5	2	0	23	27	0	0	0
	8	0	0	0	2	11	15	23	0	20	0	0	0
	9	0	0	0	0	0	7	27	20	0	23	2	25
	10	0	0	0	0	0	0	0	0	23	0	2	13
	11	0	0	0	0	0	0	0	0	2	2	0	10
	12	0	0	0	0	0	0	0	0	25	13	10	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
20	1	0	4	1	9	27	0	0	0	0	0	0	0
	2	4	0	13	10	18	0	0	0	0	0	0	0
	3	1	13	0	24	26	0	0	0	0	0	0	0
	4	9	10	24	0	15	21	3	15	0	0	0	0
	5	27	18	26	15	0	26	21	15	0	0	0	0
	6	0	0	0	21	26	0	23	17	17	0	0	0
	7	0	0	0	3	21	23	0	18	16	0	0	0
	8	0	0	0	15	15	17	18	0	22	0	0	0
	9	0	0	0	0	0	17	16	22	0	7	10	24
	10	0	0	0	0	0	0	0	0	7	0	14	5
	11	0	0	0	0	0	0	0	0	10	14	0	11
	12	0	0	0	0	0	0	0	0	24	5	11	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
21	1	0	15	24	5	2	0	0	0	0	0	0	0
	2	15	0	11	11	2	0	0	0	0	0	0	0
	3	24	11	0	17	4	0	0	0	0	0	0	0
	4	5	11	17	0	6	25	2	25	0	0	0	0
	5	2	2	4	6	0	2	5	16	0	0	0	0
	6	0	0	0	25	2	0	23	13	4	0	0	0
	7	0	0	0	2	5	23	0	21	8	0	0	0
	8	0	0	0	25	16	13	21	0	19	0	0	0
	9	0	0	0	0	0	4	8	19	0	19	19	6
	10	0	0	0	0	0	0	0	0	19	0	15	17
	11	0	0	0	0	0	0	0	0	19	15	0	9
	12	0	0	0	0	0	0	0	0	6	17	9	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
22	1	0	22	18	19	24	0	0	0	0	0	0	0
	2	22	0	1	9	17	0	0	0	0	0	0	0
	3	18	1	0	15	12	0	0	0	0	0	0	0
	4	19	9	15	0	16	8	5	21	0	0	0	0
	5	24	17	12	16	0	9	4	15	0	0	0	0
	6	0	0	0	8	9	0	16	8	13	0	0	0
	7	0	0	0	5	4	16	0	10	19	0	0	0
	8	0	0	0	21	15	8	10	0	16	0	0	0
	9	0	0	0	0	0	13	19	16	0	3	1	18
	10	0	0	0	0	0	0	0	0	3	0	21	1
	11	0	0	0	0	0	0	0	0	1	21	0	16
	12	0	0	0	0	0	0	0	0	18	1	16	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
23	1	0	6	24	8	24	0	0	0	0	0	0	0
	2	6	0	15	24	17	0	0	0	0	0	0	0
	3	24	15	0	23	3	0	0	0	0	0	0	0
	4	8	24	23	0	6	3	13	10	0	0	0	0
	5	24	17	3	6	0	7	8	27	0	0	0	0
	6	0	0	0	3	7	0	7	2	3	0	0	0
	7	0	0	0	13	8	7	0	21	12	0	0	0
	8	0	0	0	10	27	2	21	0	13	0	0	0
	9	0	0	0	0	0	3	12	13	0	9	3	18
	10	0	0	0	0	0	0	0	0	9	0	13	22
	11	0	0	0	0	0	0	0	0	3	13	0	7
	12	0	0	0	0	0	0	0	0	18	22	7	0



Вариант		1	2	3	4	5	6	7	8	9	10	11	12
24	1	0	20	9	18	23	0	0	0	0	0	0	0
	2	20	0	24	27	12	0	0	0	0	0	0	0
	3	9	24	0	16	16	0	0	0	0	0	0	0
	4	18	27	16	0	15	15	20	9	0	0	0	0
	5	23	12	16	15	0	14	11	27	0	0	0	0
	6	0	0	0	15	14	0	21	10	21	0	0	0
	7	0	0	0	20	11	21	0	8	23	0	0	0
	8	0	0	0	9	27	10	8	0	22	0	0	0
	9	0	0	0	0	0	21	23	22	0	2	2	17
	10	0	0	0	0	0	0	0	0	2	0	1	23
	11	0	0	0	0	0	0	0	0	2	1	0	27
	12	0	0	0	0	0	0	0	0	17	23	27	0

Вариант		1	2	3	4	5	6	7	8	9	10	11	12
25	1	0	26	15	5	24	0	0	0	0	0	0	0
	2	26	0	5	27	25	0	0	0	0	0	0	0
	3	15	5	0	25	18	0	0	0	0	0	0	0
	4	5	27	25	0	19	1	24	13	0	0	0	0
	5	24	25	18	19	0	22	11	27	0	0	0	0
	6	0	0	0	1	22	0	12	19	2	0	0	0
	7	0	0	0	24	11	12	0	2	23	0	0	0
	8	0	0	0	13	27	19	2	0	5	0	0	0
	9	0	0	0	0	0	2	23	5	0	17	3	24
	10	0	0	0	0	0	0	0	0	17	0	5	23
	11	0	0	0	0	0	0	0	0	3	5	0	9
	12	0	0	0	0	0	0	0	0	24	23	9	0

## **ЛАБОРАТОРНАЯ РАБОТА № 4**

### **ЭНЕРГОСИСТЕМА НА ОСНОВЕ ВЗАИМОДЕЙСТВИЯ АГЕНТОВ.**

#### **1. НАЗНАЧЕНИЕ И КРАТКАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

В процессе выполнения данной работы студенты получают навыки создания архитектур МАС с несколькими типами агентов.

#### **2. ОПИСАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ**

Данная лабораторная работа предполагает создание рынка электроэнергии, на котором присутствуют потребители ЭЭ, производители ЭЭ и распределительные компании. Каждый элемент системы представляется в виде агента, который выполняет свою функцию:

- агент потребитель, обладая знаниями о характере своей нагрузки, выдает задание на заключение контракта на снабжение агенту-поставщику исходя из своих нужд на текущий час
- агент-поставщик принимает заявки от потребителя и производит заключение контрактов на поставку ЭЭ на один час в виде открытых аукционах для получения наиболее выгодного предложения у текущих производителей ЭЭ
- агент производитель ЭЭ вырабатывает ЭЭ и участвует в торгах, стремясь продать свою доступную мощность по наиболее высокой цене

Агенты-производители представлены тремя разными видами электростанций, каждая из которых имеет уникальный график генерации:

1. ТЭС имеет постоянное значение выработки электроэнергии в час:

$$P(t) = A$$

2. ВЭС имеет функцию генерации в виде нормального распределения с мат. ожиданием  $B_1$  и отклонением  $B_2$ .

3. СЭС имеет следующую функцию генерации:

$$P(t) = \begin{cases} \sum_{i=0}^4 C_i \cdot t^i, & 5 < t < 19, \\ 0, & t < 5; t > 19; \end{cases}$$

Принять, что каждый полученный по функции график повторяется ежедневно. Коэффициенты для построения графиков  $A$ ,  $B_1$ ,  $B_2$  и  $C_i$  приведены в табл. 1.

В результате выполнения лабораторной работы должна быть реализована саморегулируемая система, где производится симуляция процесса производства, потребления и распределения ЭЭ внутри небольшой энергосистемы в рамках ускоренного 24-часового цикла.

### **3. ЗАДАНИЕ НА РАБОТУ В ЛАБОРАТОРИИ**

В рамках лабораторной работы необходимо:

5. Разработать класс агента `ConsumerAgent.class`, который выполняет следующие функции:
  - a. Исходя из заданного в конфигурационном файле графика нагрузки отправляет задание агенту-поставщику на покупку мощности
  - b. Получает отчет о результате произведения закупки от агента-поставщика
6. Разработать класс агента `DistributerAgent.class`, который выполняет следующие функции:
  - a. Принимает задание от своего агента-потребителя
  - b. Производит закупку мощности у агентов-производителей в виде открытых аукционов на снижение ставки
  - c. Отправляет отчет о выполнении задания агенту-производителю с информацией о закупленной мощности и ее цене
7. Разработать класс агента `ProducerAgent.class`, который выполняет следующие функции:
  - a. вырабатывает заданную мощность в соответствии с своими исходными данными
  - b. чувствует в торгах ЭЭ для продажи доступной мощности агентам-поставщикам
8. Подготовить тестовые сценарии запуска единичных торгов для 3-х различных сценариев
9. Провести запуск системы, состоящей из 3-х агентов-потребителей, 3-х агентов поставщиков и 3-х агентов-производителей в ускоренном в 48 раз режиме моделирования.

## 4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ

К пункту 1. Агент-потребитель должен выполнять функции отправки задания на закупку ЭЭ своему агенту-поставщику в соответствии с своим заданным в конфигурационном файле графиком нагрузки. Задание отправляется с указанием той мощности в кВт, которую необходимо закупить в рамках текущего часа, а также максимальной цены за кВтч ЭЭ. Агент также должен выполнять прием сообщений от агента-поставщика о результате произведенных закупок. В результате отчета агент-потребитель должен оценить результаты и при необходимости скорректировать цену на ЭЭ, в случае неудачной закупки, а также вывести в консоль отчет, который содержит информацию о требуемой мощности для закупки в рамках этого часа, успешной закупленной мощности и потраченных средствах. Каждый агент-потребитель связан только с одним своим агентом-поставщиком.

К пункту 2. Агент-поставщик связывает конечного потребителя и генерацию. Запрос потребителя к агенту-дистрибьютору выражается, во-первых, в количестве необходимой мощности на текущий час, а во-вторых, в максимальной цене, которую потребитель готов заплатить за электроэнергию.

Получив информацию о запросе потребителя, агент выходит на рынок услуг генерации с целью купить требуемую мощность. На рынке он организывает открытый аукцион для всех агентов-производителей, готовых передать выдвинутое агентом-дистрибьютором количество энергии. Поиск агентов-производителей необходимо производить с помощью сервиса DF. Весь цикл проведения аукциона должен представлять собой конечный автомат (FSMBehaviour).

Открытые торги должны быть представлены в виде общего topic-чата, где агенты – производители ЭЭ отправляют и получают ставки друг друга и участвует только один агент-поставщик. Агенту-производителю, предложившему наименьшую ставку, отправляется сообщение о заключении контракта в случае удовлетворения требования агента потребителя по цене. Торги должны заканчиваться по таймауту. Если ни один агент не согласился участвовать в торгах по причине отсутствия требуемого количества ЭЭ, агент дистрибьютер должен разделить контракт на несколько частей и попытаться получить ЭЭ от нескольких поставщиков. Деление контракта проводить 1 раз.

Итог проведения закупки ЭЭ агент-поставщик отправляет агенту-потребителю.

К пункту 3. Агент-производитель ЭЭ в соответствии с своим графиком нагрузки каждый час виртуального времени имеет некоторое количество свободной мощности, которую могут приобрести в конечном агенты-потребители. Агент-производитель ЭЭ участвует в торгах, организованных агентом-поставщиком и стремится продать свою свободную мощность с

наиболее выгодной для него ценой. Цена на ЭЭ от каждого производителя — это многофакторный параметр, который может определяться комплексом различных влияющих на него величин. В рамках лабораторной, однако, для упрощения можно принять, что цена ЭЭ будет зависеть только от количества вырабатываемой агентом ЭЭ — чем больше мощность, тем меньше цена. Таким образом, агент ТЭЦ будет иметь постоянную цену на всем временном отрезке, а цена от ВЭС и СЭС будет варьироваться в зависимости от текущих метеорологических условий.

Агенты-производители учувствуют в аукционах и соревнуются между собой за возможность продать ЭЭ. При этом у агента должна быть его минимальная цена, которая определяется как было описано выше, и стартовая цена — цена, с которой агент входит в торги. В лабораторной работе стоит принять, что стартовая цена равна удвоенной минимальной цене. Проведение аукциона должно быть оформлено в виде конечного автомата (FSMBehaviour).

После успешного проведения торгов и победы на аукционе агент-производитель должен резервировать мощность, запрошенную агентом-поставщиком и учитывать эту зарезервированную мощность при следующих торговых сессиях

К пункту 4. Для проверки работы комплексной системы следует провести Junit тестирование различных ситуаций вне суточного цикла. Для этого следует задать одного агента-потребителя, одного агента-поставщика и 2-3 агента-производителя, а также соответствующие параметры к ним. Параметры агентов задаются исходя из условия тестируемого сценария. Каждый сценарий должен быть оформлен в виде одного теста. В каждом сценарии воспроизводится полный цикл работы системы: потребитель отправляет запрос поставщику, поставщик организывает торги, в которых учувствуют производители, а также определяет победителя. Результат возвращается потребителю:

- Сценарий 1: Торги с единственным производителем. Задать такое количество покупаемой мощности, чтобы только 1 поставщик смог удовлетворить запросу. Ожидаемый результат: агент-производитель продает по завышенной цене мощность, однако контракт отклоняется поставщиком из-за большой цены.
- Сценарий 2: Успешный аукцион с двумя участниками. Задать такое количество покупаемой мощности, чтобы два поставщика смогли удовлетворить запросу и начали процесс снижения цены. Ожидаемый результат: агенты соревнуются друг с другом для право продать ЭЭ, и один из агентов-производителей продает по удовлетворительной цене запрошенную мощность.
- Сценарий 3: Дефицит мощности в системе. Задать такое количество покупаемой мощности, что ни один производитель не

может полностью удовлетворить запрос. Ожидаемый результат: агент дистрибьютер должен разбить контракт на несколько частей и закупить требуемое количество у различных поставщиков.

К пункту 5. Моделирование системы необходимо производить с учетом виртуального времени. Для этого необходимо создать служебный класс, в котором содержится информация о текущем состоянии моделируемого времени:

- Начинать отсчет времени со старта программы
- Создать метод, который возвращает текущий моделируемый час
- Создать метод, который возвращает время в миллисекундах до начала следующего часа

Начинать моделирование времени необходимо при старте программы.

## 5. ВАРИАНТЫ ЗАДАНИЙ

Варианты заданий представлены в таблицах ниже (Таблица 1 и Таблица 2):

Таблица 3 – исходные данные агентов

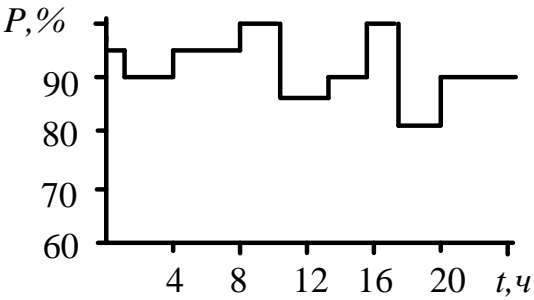
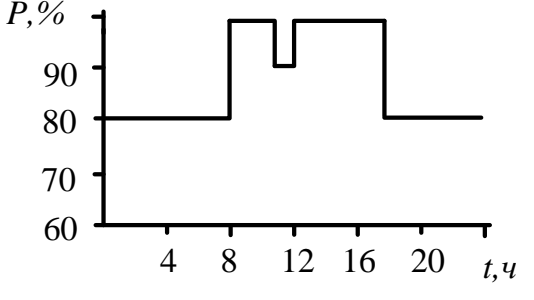
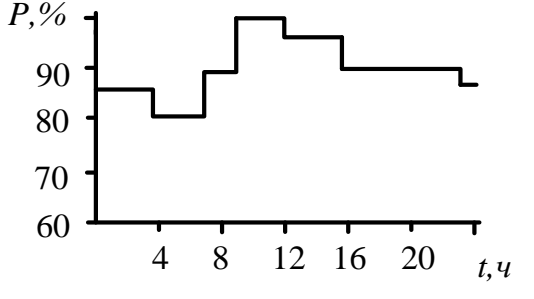
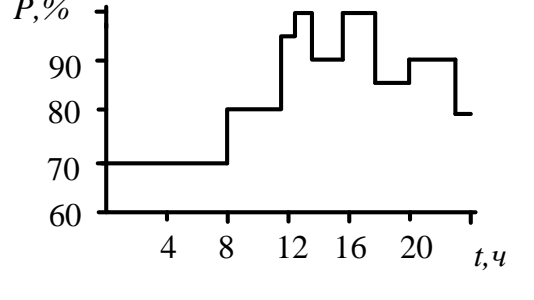
№ Варианта	A	B <sub>1</sub>	B <sub>2</sub>	C <sub>0</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	Графики нагрузок*	P <sub>100%</sub>
1	15,7	7,2	8,3	-78,985	20,313	-1,3185	0,0247	327	18,8
2	18,1	10,7	12,3	-71,087	18,282	-1,187	0,022	324	15,2
3	11,0	10,2	11,3	-75,036	19,297	-1,253	0,023	245	23,2
4	5,2	13,0	11,0	-81,355	20,922	-1,358	0,025	636	17,6
5	9,6	7,2	4,8	-86,884	22,344	-1,450	0,027	823	19,2
6	8,1	5,8	7,2	-81,355	20,922	-1,358	0,025	243	14,6
7	11,0	6,9	8,8	-77,405	19,907	-1,292	0,024	683	17,5
8	3,9	7,4	11,3	-94,782	24,376	-1,582	0,030	734	15,9
9	5,9	9,7	6,5	-89,253	22,954	-1,490	0,028	348	24,4
10	18,0	10,9	6,9	-85,304	21,938	-1,424	0,027	714	18,7
11	11,1	5,0	4,6	-78,985	20,313	-1,3185	0,0247	711	7,4
12	7,8	8,4	7,1	-71,087	18,282	-1,187	0,022	224	22,3
13	8,4	7,0	10,6	-75,036	19,297	-1,253	0,023	512	7,3
14	14,9	6,8	6,2	-81,355	20,922	-1,358	0,025	373	17,6
15	14,7	5,8	6,0	-86,884	22,344	-1,450	0,027	513	23,5
16	10,8	10,2	7,9	-81,355	20,922	-1,358	0,025	436	8,6
17	7,9	10,3	8,6	-77,405	19,907	-1,292	0,024	278	6,8
18	11,3	9,7	5,0	-94,782	24,376	-1,582	0,030	248	10,1
19	10,9	7,1	6,4	-89,253	22,954	-1,490	0,028	136	16,8
20	11,6	9,1	6,2	-85,304	21,938	-1,424	0,027	635	12,6
21	7,7	6,6	9,8	-78,985	20,313	-1,3185	0,0247	357	23,9
22	5,0	10,5	7,0	-71,087	18,282	-1,187	0,022	322	9,0
23	10,7	12,6	5,5	-75,036	19,297	-1,253	0,023	337	13,1

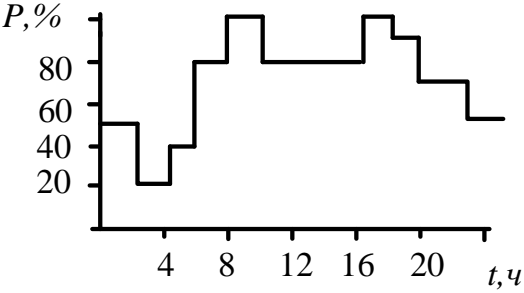
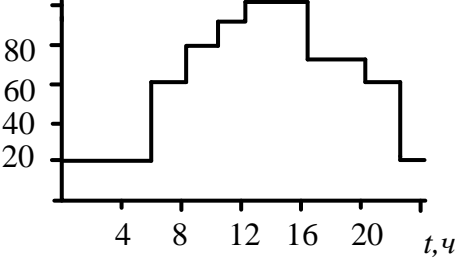
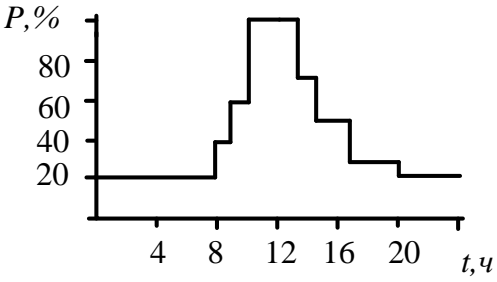
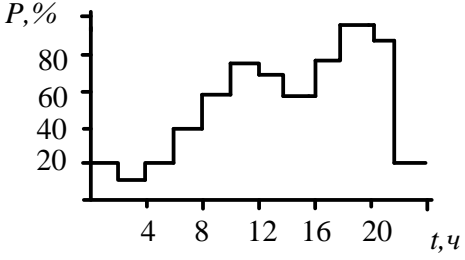
24	17,4	8,6	9,3	-81,355	20,922	-1,358	0,025	584	15,2
25	9,0	9,9	8,8	-86,884	22,344	-1,450	0,027	624	13,7
26	12,8	8,1	9,0	-81,355	20,922	-1,358	0,025	317	3,1
27	13,8	9,9	8,4	-77,405	19,907	-1,292	0,024	716	18,9
28	4,2	6,7	7,0	-94,782	24,376	-1,582	0,030	123	15,1
29	11,0	3,8	8,6	-89,253	22,954	-1,490	0,028	325	10,1
30	13,0	7,1	4,5	-85,304	21,938	-1,424	0,027	742	16,5

\*Графики нагрузок представлены в таблице 2



Таблица 4 - графики нагрузок для агентов потребителей

1. Завод ХимПром	 <p>Graph showing load <math>P, \%</math> versus time <math>t, ч</math> for Завод ХимПром. The load starts at 95%, drops to 90% at <math>t=2</math>, rises to 95% at <math>t=4</math>, drops to 88% at <math>t=10</math>, rises to 92% at <math>t=12</math>, drops to 82% at <math>t=18</math>, and rises to 90% at <math>t=20</math>.</p>
2. Завод ЦветМет	 <p>Graph showing load <math>P, \%</math> versus time <math>t, ч</math> for Завод ЦветМет. The load is constant at 80% until <math>t=8</math>, rises to 95% until <math>t=12</math>, drops to 90% until <math>t=16</math>, and returns to 80% until <math>t=20</math>.</p>
3. Завод ПищПром	 <p>Graph showing load <math>P, \%</math> versus time <math>t, ч</math> for Завод ПищПром. The load starts at 85%, drops to 80% at <math>t=2</math>, rises to 90% at <math>t=8</math>, drops to 95% at <math>t=12</math>, and returns to 90% at <math>t=16</math>, remaining constant until <math>t=20</math>.</p>
4. Обувная Фабрика	 <p>Graph showing load <math>P, \%</math> versus time <math>t, ч</math> for Обувная Фабрика. The load is constant at 70% until <math>t=8</math>, rises to 80% until <math>t=12</math>, drops to 90% until <math>t=16</math>, rises to 95% until <math>t=18</math>, drops to 85% until <math>t=20</math>, and returns to 70% until <math>t=22</math>.</p>

5.Городской транспорт	 <p>Step graph showing percentage <math>P, \%</math> on the y-axis (0 to 80+) and time <math>t, ч</math> on the x-axis (0 to 24). The data points are approximately: (0, 50), (4, 20), (8, 80), (12, 80), (16, 90), (20, 85), (24, 55).</p>
6. Фуд-корт	 <p>Step graph showing percentage <math>P, \%</math> on the y-axis (0 to 80+) and time <math>t, ч</math> on the x-axis (0 to 24). The data points are approximately: (0, 20), (6, 20), (8, 65), (10, 80), (12, 90), (16, 90), (18, 75), (20, 65), (24, 20).</p>
7. МЭИ	 <p>Step graph showing percentage <math>P, \%</math> on the y-axis (0 to 80+) and time <math>t, ч</math> on the x-axis (0 to 24). The data points are approximately: (0, 20), (8, 20), (10, 40), (12, 65), (14, 90), (16, 75), (18, 55), (20, 35), (24, 20).</p>
8. Жилое здание	 <p>Step graph showing percentage <math>P, \%</math> on the y-axis (0 to 80+) and time <math>t, ч</math> on the x-axis (0 to 24). The data points are approximately: (0, 20), (2, 10), (4, 20), (6, 40), (8, 60), (10, 75), (12, 70), (14, 60), (16, 80), (18, 90), (20, 85), (24, 20).</p>