# Модели, методы, алгоритмы и программная инфраструктура для решения сеточных уравнений в гетерогенных вычислительных системах

# 1 ЗАДАЧИ

**Задача 1**

Реализовать методы 1 и 2.

*Метод 1.* Предназначен для выбора из одномерного массива исходных данных определенного фрагмента. Мы передаем методу:

* указатель на исходный массив данных;
* параметры выборки (наследник класса GridBlock (1D, 2D или 3D);
* флаг, который указывает захватывать граничный слой или нет.

Этот метод создаёт объект класса LinearArray и возвращает этот объект.

1. Вариант 1. Реализовать последовательный алгоритм.
2. Вариант 2. Реализовать параллельный алгоритм.
3. Построить графики для сравнения при выборе массивов 1D, 2D и 3D для обоих алгоритмов.

*Метод 2.* Вернуть из массива, полученного в Методе 1 данные в большой исходный массив.

1. Вариант 1. Реализовать последовательный алгоритм.
2. Вариант 2. Реализовать параллельный алгоритм.
3. Вариант 3. Реализовать параллельный асинхронный алгоритм.
4. Построить графики для сравнения при выборе массивов 1D, 2D и 3D для обоих алгоритмов.

**Задача 2**

Реализовать метод 3.

*Метод 3.* Предназначен для обсчёта выбранного фрагмента данных.

1. Вариант 1. Реализовать алгоритм, в котором будет прямое обращение к памяти для работы с выделенным фрагментом.
2. Вариант 2. Реализовать алгоритм, в котором копируется фрагмент массива внутрь метода (локальная копия), эту копию обсчитывать и результаты возвращать в память.
3. Вариант 3. Реализовать алгоритм, в котором изменить структуру фрагмента.

**Задача 3**

Реализовать метод 4.

*Метод 4.* Предназначен для передачи граничных условий из одного фрагмента в другой.

1. Вариант 1. Реализовать алгоритм, в котором выполняется захват граничных условий: искать индексы граничных условий и передавать индексы в следующий фрагмент.
2. Вариант 2. Реализовать алгоритм, в котором копируются граничные условия из отдельного массива.
3. Вариант 3. Реализовать алгоритм, в котором граничные условия не копируются из отдельного массива, а передается указатель. Т.е., сделать граничные условия как отдельный фрагмент.

**Задача 4**

Разработать библиотеку классов, реализующую работу с массивами данных (класс LinearArray, класс LinearArrays).

**Задача 5**

Разработать потокобезопасную структуру данных для хранения набора объектов класса LinearArray (класс LinearArrays).

**Задача 6**

Разработать библиотеку классов, реализующую работу с расчётной сеткой и блоками (Grid).

**Задача 7**

Разработать потокобезопасные структуры данных для сопоставления блоков расчётной сетки с линейными массивами (с объектами LinearArrays) (классы LinearArrayToGridBlockMapping и LinearArrayToGridBlockMappings).

**Задача 8**

1. Разработать библиотеку классов и потокобезопасные структуры данных для описания структуры и технических характеристик гетерогенной распределённой вычислительной системы (Device, Nodes).
2. Разработать алгоритм тестирования производительности вычислительного устройства.

**Задача 9**

Разработать библиотеку классов и потокобезопасные структуры данных для cсопоставления блоков расчетной сетки вычислительным узлам и вычислительным устройствам (Mappings).

**Задача 10**

Подобрать оптимальный протокол передачи данных между вычислительными узлами и разработать соответствующий алгоритм для передачи данных.

1. Вариант 1. Использовать сокеты.
2. Вариант 2. Использовать протоколы.

**Задача 11**

Разработать потокобезопасные структуры данных и алгоритмы для решения СЛАУ высокой размерности итерационным МПТМ на основе графовой модели.

# 2 СПРАВОЧНАЯ ИНФОРМАЦИЯ

Формулировки из паспортов специальностей ВАК (справочная информация):

Модели, методы, алгоритмы и программная инфраструктура для организации глобально распределенной обработки данных.

Разработка, обоснование и тестирование эффективных вычислительных методов с применением современных компьютерных технологий.

Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ для проведения вычислительного эксперимента.