

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»

Институт прикладной математики и компьютерных наук
Кафедра «Прикладная математика и информатика»

Утверждено на заседании кафедры
«Прикладная математика и информатика»
« 21 » января 2021 г., протокол № 6

Заведующий кафедрой



В.И. Иванов

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (ОЦЕНОЧНЫЕ МАТЕРИАЛЫ) ДЛЯ
ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО
ДИСЦИПЛИНЕ (МОДУЛЮ)**

«Вычислительные системы и параллельная обработка данных»

**основной профессиональной образовательной программы
высшего образования – программы бакалавриата**

по направлению подготовки

01.03.02 Прикладная математика и информатика

с направленностью (профилем)

Прикладная математика и информатика

Форма обучения: очная

Идентификационный номер образовательной программы: 010302-01-21

Тула 2021 год

ЛИСТ СОГЛАСОВАНИЯ
фонда оценочных средств (оценочных материалов)

Разработчик:

Скобелцын С.А. доцент, к.ф.-м-н.

(ФИО, должность, ученая степень, ученое звание)



(подпись)

1. Описание фонда оценочных средств (оценочных материалов)

Фонд оценочных средств (оценочные материалы) включает в себя контрольные задания и (или) вопросы, которые могут быть предложены обучающемуся в рамках текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю). Указанные контрольные задания и (или) вопросы позволяют оценить достижение обучающимся планируемых результатов обучения по дисциплине (модулю), установленных в соответствующей рабочей программе дисциплины (модуля), а также сформированность компетенций, установленных в соответствующей общей характеристике основной профессиональной образовательной программы.

Полные наименования компетенций и индикаторов их достижения представлены в общей характеристике основной профессиональной образовательной программы.

2. Оценочные средства (оценочные материалы) для проведения текущего контроля успеваемости обучающихся по дисциплине (модулю)

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.1)

1. Архитектура вычислительной системы.
2. Особенности организации памяти в вычислительных системах.
3. Проблемы повышения производительности вычислений.
4. Средства повышения производительности.
5. Архитектура параллельных вычислительных систем.
6. Технология параллельного программирования OpenMP.
7. Идеология программирования на основе OpenMP.
8. Синтаксис директив в OpenMP.
9. Особенности реализации директив OpenMP.
10. Директивы загрузки и синхронизации процессов в OpenMP.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.1)

1. Параллельные вычисления линейной алгебры в технологии OpenMP.
2. Параллельная реализация методов сортировки на основе технологии OpenMP.
3. Организация многопоточных приложений в Windows API.
4. Создание процесса. Указание исполняемого модуля и командной строки.
5. Наследуемые дескрипторы. Счетчики дескрипторов процессов.
6. Идентификаторы процессов. Завершение и прекращение выполнения процесса.
7. Ожидание завершения процесса. Защита процесса.
8. Управление потоками в Windows.
9. Идентификация потоков.
10. Функции управления потоками.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.2)

1. Приостановка и возобновление выполнения потока.
2. Ожидание завершения потока в Windows.
3. Удаленные потоки. Использование библиотеки C в потоках.
4. Библиотеки с многопоточной поддержкой.
5. Потоки и производительность.

6. Модель "хозяин/рабочий" и другие модели многопоточных приложений.
7. Моделирование управления процессами в Windows API.
8. Моделирование управления потоками в Windows API.
9. Средства организации многопоточных приложений в .NET.
10. Библиотек TPL. Создание процесса функциями C#.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.2)

1. Класс Parallel и его методы. Понятие задачи - Task.
2. Формирование пакета задач Parallel.Invoke.
3. Создание множества задач методом Parallel.For.
4. Средства синхронизации в .NET.TPL.
5. Параллельное программирование с использованием технологии MPI.
6. Параллельные программы на основе передачи сообщений.
7. Параллельные процессы, взаимодействующие с помощью передачи сообщений.
8. Пересылка данных между двумя процессами.
9. Численное интегрирование: параллельная реализация на основе MPI.
10. Семантика точечных обменов.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.3)

1. Организация буферизованных пересылок.
2. Параллельные вычисления линейной алгебры в технологии MPI.
3. Реализация расчетов по квадратурным формулам на основе технологии MPI.
4. Программирование массивно-параллельных вычислительных систем на основе технологии CUDA.
5. Основы GPU-вычислений в технологии CUDA.
6. Развитие GPU-вычислений. Технология CUDA.
7. Средства разработки в CUDA.
8. Особенности реализации CUDA C.
9. Параллельное программирование в CUDA.
10. Параллельные вычисления линейной алгебры на GPU.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.3)

1. Параллельная реализация методов сортировки на основе технологии CUDA.
2. Средства параллельного программирования Java.
3. Потоки в Java. Модель потоков Java.
4. Приоритеты потоков. Синхронизация.
5. Обмен сообщениями в Java.
6. Java: класс Thread и интерфейс Runnable.
7. Java: главный поток. Создание потока.
8. Реализация Runnable в Java.
9. Расширение Thread в Java.
10. Создание множества потоков в Java.

3. Оценочные средства (оценочные материалы) для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.1)

1. Элементы теории параллельного программирования.
2. Модель графа алгоритма параллельных вычислений.
3. Внутренний параллелизм.
4. Графовые модели программ.
5. Графы зависимостей и минимальные графы.
6. Построение минимальных графов зависимостей.
7. Циклы ParDO и избыточные вычисления.
8. Развертки графа. Макрографы зависимостей.
9. Концепция GRID и метакомпьютинг.
10. Векторно-конвейерные компьютеры.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.1)

1. Средства параллельного программирования Java.
2. Потоки в Java. Модель потоков Java.
3. Приоритеты потоков. Синхронизация.
4. Обмен сообщениями в Java.
5. Java: класс Thread и интерфейс Runnable.
6. Java: главный поток. Создание потока.
7. Реализация Runnable в Java.
8. Расширение Thread в Java:.
9. Создание множества потоков в Java.
10. Java: использование isAlive() и join().

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.2)

1. Законы Амдала и Густавсона-Барсиса.
2. Производительность параллельных ВС.
3. Современные направления развития параллельных вычислительных систем.
4. Основные принципы технологии параллельного программирования OpenMP.
5. Основные директивы OpenMP, примеры.
6. Переменные окружения OpenMP, примеры.
7. Библиотечные функции OpenMP, примеры.
8. Варианты синхронизация потоков в OpenMP (atomic critical barrier master и др.).
9. Концепции и составляющие библиотеки PFX.
10. Класс Parallel. Синтаксис и особенности использования методов Parallel.For, Parallel.ForEach.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.2)

1. Понятие LINQ-запросов. Синтаксис запросов и синтаксис методов в LINQ.
2. Автоматическое распараллеливание запросов в PLINQ.
3. Общие принципы параллельного программирования основе передачи сообщений (технологии MPI).
4. Параллельные процессы, взаимодействующие с помощью передачи сообщений.
5. MPI: пересылка данных между двумя процессами.

6. Библиотечные функции в MPI, примеры.
7. Алгоритм параллельной реализации в MPI численного интегрирования.
8. Правила и средства точечных обменов в MPI.
9. Организация буферизованных пересылок в MPI.
10. Основные идеи GPU-вычислений.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-3 (контролируемый индикатор достижения компетенции ПК-3.3)

1. Технология CUDA: взаимодействие с хост-системой.
2. Программная архитектура CUDA.
3. Взаимодействие потоков (нитей) в технологии CUDA.
4. Типы памяти в архитектуре CUDA.
5. Найти максимальный элемент в одномерном массиве чисел типа float, используя многопоточную обработку.
6. Найти минимальный элемент в одномерном массиве целых чисел, используя многопоточную обработку.
7. Для заданного массива A чисел типа double получить новый - B такой, чтобы $B[i]=2*A[i]-1$, используя многопоточную обработку.
8. Найти номер строки и столбца максимального элемента в двумерном массиве чисел типа float, используя многопоточную обработку.
9. Найти номер максимального элемента в одномерном массиве чисел типа float, используя многопоточную обработку.
10. Найти максимальный элемент в двумерном массиве чисел типа float, используя многопоточную обработку.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ПК-5 (контролируемый индикатор достижения компетенции ПК-5.3)

1. Найти частичную сумму N членов гармонического ряда, используя многопоточную обработку.
2. Найти частичную сумму N членов числовой последовательности с общим членом $n/(n+1)$, используя многопоточную обработку.
3. Найти частичную сумму N членов числовой последовательности с общим членом $(n-3)/(n+2)$, используя многопоточную обработку.
4. На отрезке $[a,b]$ при заданном количестве разбиений N найти значение интеграла $f(x)=1-2*\sin(x)$ с помощью формулы прямоугольников, используя многопоточную обработку.
5. На отрезке $[a,b]$ при заданном количестве разбиений N найти значение интеграла $f(x)=1+x-2*x*x$ с помощью формулы трапеции, используя многопоточную обработку.
6. На отрезке $[a,b]$ при заданном количестве разбиений N найти значение аргумента дающего максимальное значение функции $f(x)$, используя многопоточную обработку.
7. На отрезке $[a,b]$ при заданном количестве разбиений N найти максимальное значение функции $f(x)$, используя многопоточную обработку.
8. Найти скалярное произведение векторов, представленных одномерными массивами с числами типа double (заполнение массивов - псевдослучайные числа из $[0,1]$), используя многопоточную обработку.
9. Найти максимальный элемент из поэлементной суммы двух двумерных массивов с числами типа float (заполнение массивов - псевдослучайные числа из $[0,100]$), используя многопоточную обработку.
10. Найти максимальный элемент в одномерном массиве чисел типа float, используя многопоточную обработку.