

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Ингушский государственный университет»**



УТВЕРЖДАЮ

Проректор по учебной работе

З.О.Батыгов

05 2018 г.

**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ПРАКТИКИ**

(практика по получению первичных профессиональных умений и навыков)

**Основная профессиональная образовательная программа
академического бакалавриата**

03.03.02 – Физика

Квалификация выпускника

Бакалавр

Форма обучения

очная

МАГАС, 2018

Учебная (вычислительная) практика

Вычислительная практика студентов проводится согласно учебному плану направления во 4 семестре в компьютерных классах физико-математического факультета.

Базовые курсы информатики в сочетании с учебной вычислительной практикой на персональном компьютере позволяют расширить возможности подготовки студентов к широкому использованию компьютерных технологий в будущей практической деятельности, предоставляют возможность развития и углубления знаний, умений и навыков студентов в соответствующей предметной области, а также способствуют:

закреплению теоретического материала, изложенного на лекциях дисциплин предметного блока;

приобретению практических навыков по разработке и реализации алгоритмов различной природы, использованию современных систем программирования в профессиональной деятельности;

повторению теоретического материала, необходимого для решения задач;

раскрытию связей между фундаментальными науками и задачами практического содержания;

развитию логического мышления студентов, повышению уровня их математической и вычислительной культуры.

ЦЕЛИ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ

Основная цель вычислительной практики состоит:

в обучении студентов приемам и методам разработки надежных программ;

в привитии навыков программирования, создания наглядных программ и их отладки;

в обучении приемам и методам тестирования программ.

ЗАДАЧИ УЧЕБНО-ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ.

Студенты на вычислительной практике должны:

закрепить полученные в ходе изучения базовых компьютерных дисциплин знания и умения по методам использования, разработки и реализации классических алгоритмов на изученных языках программирования;

изучить требования к оформлению отчетов, методы и прикладные среды для подготовки отчетов по результатам практики;

изучить и соблюдать технику безопасности условий работы программиста;

ориентироваться в технических характеристиках используемой вычислительной техники;

освоить методы профессионального поиска в сети Internet информации по современным методам программирования и выработки навыков обеспечения потребностей будущей профессиональной деятельности.

Практические навыки разработки программ должны приобретаться студентом путем выполнения индивидуального задания по согласованию с руководителем практики.

ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ:

В результате прохождения учебно-вычислительной практики обучающийся должен приобрести следующие практические навыки, умения, универсальные и профессиональные компетенции:

способностью использовать основные методы, способы и средства получения, хранения, переработки информации и навыки работы с компьютером как со средством управления информацией (ОПК-5);

способностью использовать специализированные знания в области физики для освоения профильных физических дисциплин (ПК-1);

способностью проводить научные исследования в избранной области экспериментальных и (или) теоретических физических исследований с помощью современной

приборной базы (в том числе сложного физического оборудования) и информационных технологий с учетом отечественного и зарубежного опыта (ПК-2);

научно-инновационная деятельность:

готовностью применять на практике профессиональные знания теории и методов физических исследований (ПК-3);

способностью применять на практике профессиональные знания и умения, полученные при освоении профильных физических дисциплин (ПК-4);

способностью пользоваться современными методами обработки, анализа и синтеза физической информации в избранной области физических исследований (ПК-5);

организационно-управленческая деятельность:

способностью понимать и использовать на практике теоретические основы организации и планирования физических исследований (ПК-6);

способностью участвовать в подготовке и составлении научной документации по установленной форме (ПК-7);

СОДЕРЖАНИЕ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ

Основными видами деятельности студента при прохождении учебной вычислительной практики являются:

1. Изучение учебного и учебно-методического материала по предложенной теме практики.
2. Изучение необходимых для решения поставленной задачи программных средств.
3. Изучение методов проектирования и реализации интерфейсов разрабатываемых программ.
4. Проектирование комплекса программ для выполнения поставленной задачи.
5. Разработка программной реализации задачи с помощью выбранного языка программирования.
6. Разработка тестовых заданий по проверке работоспособности программ.
7. Проведение контрольных проверок по работе комплекса программ.
8. Подготовка технической документации по разработанному комплексу программ.
9. Подготовка отчетной документации по результатам прохождения практики.

Вычислительная практика рассчитана на студентов 2 курса физико-математического факультета (направления – «Физика»). Задания, приведенные в сборнике, разбиты на следующие разделы: использование оператора выбора, цикла с параметром, обработка одномерных и двумерных массивов, обработка строк.

Все они требуют предварительного знания не только основ программирования, но и чисто математических дисциплин, таких как: алгебра, аналитическая геометрия. Однако, для успешного написания программ по предлагаемым задачам достаточно знаний первого семестра первого курса, что дает возможность проводить ВП, начиная уже со второго семестра первого курса. При прохождении вычислительной практики студент получает задания из каждого раздела сборника заданий по учебно-вычислительной практике, разработанного сотрудниками кафедры.

ОРГАНИЗАЦИЯ ВП

Организация вычислительной практики предполагает обязательное регулярное проведение консультаций, проводимых в виде аудиторных занятий, сочетающих как индивидуальную форму работы преподавателя со студентами, так, и коллективную, связанную с изложением преподавателем указаний и рекомендаций по выполнению заданий, а также с разбором типичных ошибок, допускаемых студентами в процессе выполнения заданий.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Методические указания регламентируют процесс разработки программной системы, предусмотренной заданием. Процесс разработки программы, реализующей решение

предложенной в задании задачи, необходимо разбить на этапы в соответствии с технологическим циклом разработки программного обеспечения: постановка задачи, анализ, проектирование, кодирование, тестирование и сопровождение. Прохождение данных этапов необходимо документировать, то есть одновременно с выполнением работ очередного этапа готовить соответствующий раздел отчета по учебной вычислительной практике.

1. Постановка задачи. В контексте вычислительной практики на данном этапе необходимо ознакомиться со своим заданием, определить перечень возможных задач, необходимых для выполнения задания и выяснить все неясные моменты задания у преподавателя, которой руководит вычислительной практикой данного студента.

2. Анализ. В контексте вычислительной практики на данном этапе необходимо подготовить раздел отчета «Анализ». В данном разделе отчета формулируется задание, определяются отдельные функции будущей программы, описываются форматы входных и выходных данных и приводятся примеры входных и соответствующих им выходных данных. Полученные результаты обсуждаются в ходе консультации с преподавателем.

3. Проектирование. На этапе проектирования необходимо выполнить проектирование структуры программы, разработать соответствующие алгоритмы, разработать набор тестов. Важной особенностью этапа проектирования является то, что все работы на данном этапе выполняются без использования системы программирования.

Прежде всего, в зависимости от поставленной задачи, надо определиться с выбором подхода проектирования, разработать заголовки и спецификации процедур и функций (модулей) программы, а также соответствующие структуры данных, разработать текст головного модуля программы.

Головной модуль должен содержать корректные вызовы модулей первого уровня иерархии. В ходе работ на данном этапе необходимо подготовить раздел отчета «Проектирование».

Далее следует разработать алгоритмы реализации модулей. После этого необходимо разработать тесты для каждого модуля, входящего в модульную структуру (в том числе для головного модуля). Количество тестов зависит от того, насколько полно в тестовом наборе представлены возможные входные данные модуля (тесты должны проверять работу модуля на обычных и граничных данных). Качество тестов зависит от того, насколько высока вероятность обнаружения с их помощью ошибок в данном модуле.

Подготовленные в ходе работ на данном этапе наборы тестов позже войдут в раздел отчета «Тестирование».

Кодирование. На этапе кодирования разработанный ранее на бумаге алгоритм программы должен быть реализован в виде программного кода.

После того, как разработана структура текста программы, необходимо подготовить файлы для исходных текстов головного модуля программы, всех модулей и тестовых программ. В ходе работ на данном этапе необходимо подготовить раздел отчета «Кодирование».

5. Тестирование. В соответствии с планом проведения тестирования проводится проверка всех модулей программной системы (автономное тестирование) и системы в целом (комплексное тестирование). Результаты тестирования фиксируются в разделе отчета «Тестирование».

6. Сопровождение. В контексте учебно-вычислительной практики данный этап подразумевает подготовку к защите отчета о прохождении практики. К защите отчета студент обязан подготовить полностью готовый текст отчета:

1. Титульный лист;
2. Перечень условных обозначений;
3. Постановка задачи (условие);
4. Соглашение о требованиях к программе:
 - Назначение программы;

- Перечень функций, выполняемых программой;
 - Формат входных данных;
 - Формат выходных данных;
5. Алгоритм решения задачи.
 6. Программный код (листинг программы).
 7. Описание набора тестов. Результаты ручного счета тестового примера. Общий результат компьютерного тестирования.
 8. Литература.

Общая трудоемкость дисциплины.

3 зачетных единиц (108 академических часов)