

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГБОУ ВО «СГУ имени Н.Г. Чернышевского»
Факультет nano- и биомедицинских технологий

УТВЕРЖДАЮ:

Проректор по учебно-методической
работе профессор Е. Г. Елина



2016 г.

Рабочая программа дисциплины
«Принципы расширения стандартных прикладных программ»

Направление подготовки 11.03.04
«Электроника и нанoeлектроника»

Профили подготовки:

«Приборы микро- и нанoeлектроники, методы измерения микро- и наносистем»,
«Физика и технология твердотельных электронных микро- и наноструктур»,
«Микро- и нанoeлектроника, диагностика nano- и биомедицинских систем»

Квалификация (степень) выпускника

Бакалавр

Форма обучения

очная

Саратов, 2016

1 Цели освоения дисциплины

Целью освоения дисциплины «Принципы расширения стандартных прикладных программ» является формирование у студентов общепрофессиональных и специальных знаний в области прикладного программирования, в том числе в области написания, тестирования, расширения и изменения ранее написанных прикладных программ.

Задачами освоения дисциплины являются:

- формирование *знаний* об основных принципах написания, расширения, тестирования и модификации прикладных программ;
- формирование *умений* самостоятельно писать, расширять, модифицировать прикладные компьютерные программы;
- формирование *владений* языками программирования, средами разработки программ, алгоритмическими приёмами и подходами к тестирования и выявлению ошибок в программах.

2 Место дисциплины в структуре ООП бакалавриата

Профессиональная дисциплина «Принципы расширения стандартных прикладных программ» относится к вариативной части блока «Дисциплины» и изучается студентами дневного отделения факультета нано- и биомедицинских технологий СГУ, обучающимися по направлению 11.03.04 в течение 1-го учебного семестра (1-ый курс). Дисциплина опирается на базовые знания математики и информатики, доступные учащимся в результате освоения школьной программы. Дисциплина готовит студентов к освоению дисциплины базовой части «Информатика: средства и методы защиты информации» и дисциплины вариативной части «Компьютерная графика» и итоговой государственной аттестации и дисциплин по выбору: «Компьютерное моделирование, расчёт и проектирование микро- и наносистем» и «Основы математического моделирования в твердотельной электронике».

3 Компетенции обучающегося, формируемые в результате освоения дисциплины

Освоение дисциплины формирует следующие компетенции:

- ОПК-7 — способность учитывать современные тенденции развития электроники, измерительной и вычислительной техники, информационных технологий в своей профессиональной деятельности;
- ОПК-9 — способность использовать навыки работы с компьютером, владеть методами информационных технологий, соблюдать основные требования информационной безопасности.

В результате освоения дисциплины обучающийся студент должен:

- *знать* синтаксис, семантику, основные управляющие конструкции языка программирования Pascal или языка Python, ;
- *уметь* самостоятельно анализировать, расширять и тестировать программный код, при необходимости — писать новый;
- *владеть* алгоритмическими приёмами.

4 Структура и содержание дисциплины

Общая трудоёмкость дисциплины составляет 3 зачётные единицы (108 часов).

4.1 Распределение часов на по темам

№ п/п	Наименование раздела, подраздела, темы лекции	Семестр	Неделя семестра	Бюджет учебного времени					Формы текущего контроля успеваемости и промежуточной аттестации (по семестрам)
				Всего	В том числе				
					Лек- ции	Лабо- ра- тор- ные	Семи- нар- ские	СРС	
1	2			3	4	5	6	7	8
Очная полная программа									
1	Линейные программы	1	1– 2	12	2	4		6	индивидуальные отчёты по заданиям

2	Условия и логические выражения	1	3–4	12	2	4		6	индивидуальные отчёты по заданиям
3	Цикл со счётчиком	1	5–6	12	2	4		6	индивидуальные отчёты по заданиям
4	Цикл с условием	1	7–8	12	2	4		6	индивидуальные отчёты по заданиям
5	Чтение и запись текстовых файлов	1	9–10	14	2	4		8	индивидуальные отчёты по заданиям, контрольная работа
6	Одномерные массивы	1	11–12	14	2	4		8	индивидуальные отчёты по заданиям
7	Типы, записи, стандартные модули	1	13–14	14	2	4		8	индивидуальные отчёты по заданиям
8	Строки и символы	1	15–17	18	3	6		9	индивидуальные отчёты по заданиям
Аттестация по итогам семестра									зачёт
Итого по дисциплине				108	17	34		57	

4.2 Содержание дисциплины

1. Линейные программы. Структура программы. Арифметические операторы и оператор присваивания. Стандартные функции с одним аргументом и без аргумента.
2. Условный оператор, строение составных операторов. Операторы сравнения. Логические операторы и сложные инструкции, таблицы истинности. Правила форма-

тирования кода и хорошего стиля.

3. Цикл со счётчиком. Правила работы со счётчиком. Цикл с уменьшением счётчика. Досрочное завершение и пропуск итерации. Приёмы накопления суммы и произведения. Рекуррентные соотношения.
4. Цикл с условием. Переменные–флаги. Переменные–счётчики событий. Имитация цикла со счётчиком. Имитация условного оператора. Вычисления с заданной точностью. Константы.
5. Текстовые файлы. Открытие на чтение, чтение одной или нескольких записей. Переход на новую строку. Вычисления количества строк в файле. Чтение чисел, записанных в один или несколько (заранее известно сколько) столбцов. Запись в файл одного числа. Запись одного или нескольких массивов одинаковой длины. Поиск среднего, минимума и максимума из чисел в файле.
6. Понятие составных типов данных. Массивы чисел, размещение в памяти. Обращение по индексу. Стандартные функции над массивами. Копирование элементов и копирование указателя. Запись массива в файл и чтение из файла. Поиск среднего, минимума и максимума в массиве.
7. Производные типы. Записи. Стандартная библиотека: основные процедуры и функции работы с числовыми данными.
8. Символы и строки. Обычные и специальные символы. Понятие кодировки: получение символа по коду и кода по символу. Виды строк. Функции со строками.

5 Образовательные технологии

При реализации различных видов учебной работы по данной дисциплине (лекции, лабораторные работы, самостоятельная работа) с целью создания условий для самоактуализации и самореализации обучающихся, предоставления возможностей для конструирования собственного знания, используются следующие современные образовательные технологии:

- информационно-коммуникационные технологии;
- конкурсное и соревновательное обучение;
- работа в малых группах;

- творческие задания.

Аудиторные занятия проводятся в форме лекций и лабораторных работ в компьютерном практикуме. Самостоятельная работа предполагает прежде всего самостоятельное выполнение студентом заданий компьютерного практикума, а также разбор лекционного материала и материала семинарских занятий.

Курс построен таким образом, что лекционные занятия и практикум тесно связаны друг с другом, лекционный материал подготавливает студента к выполнению соответствующего практического задания. Поскольку лекции носят практическую направленность, на них, часто в интерактивной форме, объясняются в том числе различные технические аспекты практических заданий.

Лекционные занятия проводятся частично при помощи мультимедийного проектора. Перед занятиями или непосредственно после них студентам может выдаваться конспективный материал.

Практикум является главной формой занятий, т. к. развивает у студентов непосредственные навыки и служит закреплению полученного на лекциях и семинарах материала. Задания практикума выдаются с небольшим опережением, чтобы дать возможность студенту самому планировать свою работу. В аудитории студенты должны отлаживать критически важные фрагменты программ, обсуждать с преподавателем структуру и ключевые особенности реализации алгоритма, сдавать сделанные задания. Несмотря на большой объём аудиторных занятий, выполнение всех заданий в аудитории невозможно, поэтому наиболее простая и трудоёмкая часть заданий, не требующая постоянного контроля и содействия преподавателя, должна выполняться студентами в рамках самостоятельной работы.

Задания практикума предполагают наличие у студентов базовых навыков программирования и логического мышления, поскольку все задания выполняются с помощью численных методов на компьютере. Тем не менее, практикум не ставит целью расширение знаний студента в области программирования. В качестве инструментария может применяться любой знакомый студенту язык программирования, может использоваться любая операционная система. Задания практикума не предполагают наличие в языке программирования мощных встроенных библиотек численных алгоритмов и построены таким образом, чтобы следующие задания опирались на предыдущие. В качестве средства реализации может быть выбран любой современный распространённый язык, например, Pascal, Fortran, Python, C или иной. Выбор средств разработки: языка и среды является свободным для студента при соблюдении требования о технической и юридической возможности и правомочности использования выбранных средств.

5.1 Перечень лабораторных работ (примерный)

1. Арифметические выражения и стандартные функции.

Цель работы: научиться использовать арифметические операторы и стандартные функции.

Задание: используя арифметические операторы, а также стандартные функции округления, отсечения целой части и модуля, составьте арифметические выражения и напишите программу для вычисления и вывода ряда величин (необходимая информация запрашивается у пользователя).

2. Операции с несколькими числами.

Цель работы: научиться писать программы и обмениваться информацией с компьютером.

Задание: реализовать вычисление по заданной формуле функции нескольких переменных.

3. Условия и сравнения.

Цель работы: научиться использовать условный оператор и операторы сравнения.

Задание: реализовать вычисления с ветвлением в зависимости от значения введённого аргумента, конкретное задание выдаётся преподавателем.

4. Сложные условия.

Цель работы: научиться составлять сложные условия.

Задание: составить сложные условия с использованием операторов сравнения и логических операторов.

5. Повторяющиеся операции.

Цель работы: научиться реализовывать повторяющиеся операции с помощью цикла.

Задание: реализуйте вывод многократно повторяющихся символов на экран по заданному преподавателем правилу.

6. Накопление произведения и суммы.

Цель работы: овладеть алгоритмическими приёмами накопления произведения и суммы.

Задание: написать программу для суммирования или мультиплицирования с накоплением заданной последовательности, конкретный вид последовательности определяется преподавателем.

7. Рекуррентные соотношения.
Цель работы: освоить рекуррентные соотношения.
Задание: написать программу, реализующую вычисления по заданному преподавателем рекуррентному соотношению.
8. Табулирование функций.
Цель работы: научиться табулировать функции.
Задание: написать программу, которая выводит значения аргумента и функции при:
- (a) шаге по аргументу, его начальном значении и числе значений;
 - (b) шаге по аргументу, его начальном и конечном значении;
 - (c) начальном и конечном значении аргумента и числе значений.
9. Вычисление номера шага.
Цель работы: научиться вычислять номер шага, на котором происходит некоторое событие.
Задание: написать программу, которая при помощи конструкции цикла вычисляет номер шага, при котором впервые выполняется заданное преподавателем условие.
10. Вычисления с заданной точностью.
Цель работы: научиться вычислять значения по рекуррентным формулам с заданной точностью.
Задание: написать программу, реализующую вычисление по заданной рекуррентной формуле с остановкой не по числу шагов, а по достижении заданной точности.
11. Чтение из файла.
Цель работы: научиться читать из файла числовые данные, записанные в столбец или несколько столбцов.
Задание: написать программу, читающую из данного преподавателем текстового файла данные, записанные в один или несколько (заранее известное количество) столбцов.
12. Запись в текстовый файл.
Цель работы: научиться записывать числовые данные в текстовый файл в один или несколько столбцов.
Задание: написать программу, реализующую запись значений, генерируемых в цикле по оговоренному правилу, в файл.

13. Вычисление минимума и максимума, суммы, среднего значения по данным из файла.
Цель работы: научиться рассчитывать минимум, максимум сумму или среднее значения данных, записанных в файл.
Задание: написать программу, реализующую чтение числовых данных из файла, записанных в один столбец, и расчёт по ним суммы, среднего, минимального или максимального значения.
14. Чтение массива из файла.
Цель работы: научиться читать из файла числовые данные в массив.
Задание: написать программу, читающую из данного преподавателем текстового файла данные в массив.
15. Запись массива в текстовый файл.
Цель работы: научиться записывать числовые данные из массива в текстовый файл.
Задание: написать программу, реализующую запись значений одного или нескольких массивов в файл.
16. Вычисление минимума и максимума, суммы, среднего значения массива числовых данных.
Цель работы: научиться рассчитывать минимум, максимум сумму или среднее значения для массива.
Задание: написать программу, реализующую расчёт суммы, среднего, минимального или максимального значения для массива.
17. Записи.
Цель работы: освоить работу с записями (структурами).
Задание: написать программу, производящую вычисления с различными полями двух или нескольких записей, задание детализируется преподавателем.
18. Строки.
Цель работы: освоить базовые операции со строками.
Задание: написать программу, реализующую некоторые базовые операции со строками, в том числе со строками, являющимися полями записей.

5.2 Условия обучения инвалидов и лиц с ограниченными возможностями здоровья

Для обучения инвалидов и лиц с ограниченными возможностями здоровья создаются следующие дополнительные условия:

- предоставление инвалидам по зрению или слабовидящим возможностей использовать крупноформатные наглядные материалы;
- организация коллективных занятий в студенческих группах с целью оказания помощи в получении информации инвалидам и лицам с ограниченными возможностями по здоровью;
- проведение индивидуальных коррекционных консультаций для инвалидов и лиц с ограниченными возможностями здоровья;
- использование индивидуальных графиков обучения;
- использование дистанционных образовательных технологий.

6 Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

6.1 Организация самостоятельной работы

Конспекты лекций и задания для практикума выдаются студентам свободно после каждой лекции. При необходимости, они также могут получить программное обеспечение, лицензия которого позволяет его свободное распространение, в частности интерпретатор языка Python и модули к нему, среду разработки Lazarus для программирования на языке FreePascal, среду разработки Code::Blocks и набор компиляторов gcc и другие средства, которые могут быть доступны в компьютерном классе или иметься на руках у преподавателя.

Основным критерием освоения студентами курса выступает успешное выполнение ими работ лабораторного практикума. Все задания практикума строятся таким образом, чтобы как на завершающем, так и, по возможности, на промежуточных этапах

можно было проконтролировать правильность выполнения. Для контроля правильности выполнения используются специальные тестовые примеры, для которых результат известен из теоретических соображений. Данные для этих примеров могут даваться преподавателем либо, в ряде случаев, генерироваться самими студентами. В ряде случаев действенным методом контроля правильности полученных результатов является использование уже отлаженных ранее программ, например, рассчитав сумму чисел, записанных в текстовый файл, студент может использовать это результат для проверки корректности работы с двоичным.

6.2 Методические рекомендации студентам

- Систематически посещать лекции, а в случае пропуска занятий по уважительной причине в кратчайшие сроки ознакомиться с пропущенным материалом. Это позволит без лишнего напряжения и полно воспринять представленный материал, а также при выполнении практических заданий не потерять связь с лекционным курсом.
- Иметь при себе тетрадь с лекциями по курсу и на каждом занятии в лабораторном практикуме.
- Стараться выполнить наиболее сложные задания в присутствии преподавателя, оставив для самостоятельной разработки более трудоёмкие, но алгоритмически несложные части работы.
- При возникновении сложно преодолимых трудностей не стесняться консультироваться у преподавателя и обращаться к рекомендованной литературе.

6.3 Методические рекомендации преподавателю

- Ознакомить студентов с программой курса, списком экзаменационных и контрольных вопросов, списком рекомендованной литературы и планом работы на семестр на первой лекции.
- Раздать студентам материалы для самостоятельной подготовки, в том числе и задания для вычислительной практики.
- Консультировать студентов по мере прослушивания курса и выполнения ими заданий самостоятельной подготовки.

- Во время лабораторных занятий стараться охватить наиболее сложные и неочевидные моменты.

6.4 Рекомендации по использованию информационных технологий

Кроме традиционного использования учебников, монографий и периодических научных изданий для более глубокого усвоения материала дисциплины целесообразно пользоваться ресурсами интернета и электронными учебными материалами, распространяемыми на компакт-дисках и пр.

Также приветствуется использование сетевых баз данных, библиотек методов и эталонных сигналов и прочих ресурсов, содержащих полезную информацию, способную расширить и углубить знания студентов в области обработки экспериментальных рядов.

6.5 Вопросы для проведения промежуточной аттестации по итогам освоения дисциплины (в форме зачёта)

1. Линейные программы

- Понятие алгоритма, начальное представление об устройстве компьютера, устройство памяти
- Переменные, раздел объявления переменных, переменные числовых типов
- Что такое оператор, оператор присваивания, арифметические операторы
- Что такое стандартные функции, основные стандартные функции с одним аргументом и без аргумента
- Ввод и вывод в текстовом режиме

2. Логические выражения и условный оператор

- Переменные логического типа, операторы сравнения
- Логические операторы, комбинирование условий
- Условный оператор, понятие составного оператора, операторные скобки

3. Циклы

- Цикл с параметром (счётчиком): синтаксис и правила работы со счётчиком

- (b) Примеры накопления суммы, произведения
- (c) Цикл с `downto`, оператор `break`
- (d) Рекуррентные соотношения
- (e) Перебор вариантов
- (f) Переменные-флаги
- (g) Переменные счётчики событий
- (h) Цикл с условием `while`, эмитация `for`
- (i) Зацикливание, выход из цикла посредством `break`
- (j) Вычисление номера шага, вычисление с заданной точностью

4. Массивы

- (a) Объявление массива, операторы `length`, `high`, `low`
- (b) Базовые операции с массивами, вычисление индексов
- (c) Примеры программ: вычисление минимума, максимума, суммы

5. Текстовые файлы

- (a) Открытие текстового файла на чтение, чтение одного или нескольких чисел
- (b) Чтение из текстового файла одного массива известной длины
- (c) Подсчёт строк в текстовом файле, чтение массива неизвестной длины
- (d) Открытие текстового файла на запись, запись нескольких чисел в файл
- (e) Запись в файл одного массива в столбик

6. Символы и строки

- (a) Символьный тип, способы задания символьных переменных, стандартные функции с символами
- (b) Строковый тип, длина строки, чтение и запись, сложение строк
- (c) Стандартные функции со строками
- (d) Считывание и запись строк в текстовые файлы
- (e) Считывание и запись символов в текстовые файлы, функция `eofn`

7. Типы и записи

- (a) Константы
- (b) Объявление нового типа данных, типы — массивы
- (c) Записи

7 Данные для учета успеваемости студентов в БАРС

Таблица 1— максимальные баллы по видам учебной деятельности в 1 и 2 семестрах							
1	2	3	4	5	6	7	8
Лекции	Лабораторные занятия	Практические занятия	СМС	Автоматизированное тестирование	Другие виды учебной деятельности	Промежуточная аттестация	Итого
0	40	0	20	0	10	30	100

Примечание: поскольку посещение лекций в 1 семестре не оценивается, используется единая таблица для оценки деятельности.

7.1 Программа оценивания учебной деятельности студента

Лекции — посещаемость не оценивается.

Лабораторные занятия — выполнения лабораторных работ предусмотренных рабочей программой: от 0 до 40 баллов.

Практические занятия — учебным планом по данной дисциплине не предусмотрены.

Самостоятельная работа — выполнение заданий лабораторных работ самостоятельно: от 0 до 20 баллов.

Автоматизированное тестирование — не предусмотрено рабочей программой.

Другие виды учебной деятельности — выполнение индивидуальных заданий по инициативе студента.

Промежуточная аттестация (зачёт) — в зависимости от ответа студента на зачёте: от 0 до 30 баллов.

Подготовка студента к прохождению промежуточной аттестации осуществляется в ходе лекционных и лабораторных занятий, а также во внеаудиторные часы в рамках самостоятельной работы студента. Во время самостоятельной подготовки студент пользуется конспектами лекций, основной и дополнительной литературой по дисциплине. Промежуточная аттестация проводится в виде письменного зачёта. Во время проведения зачёта студент должен дать развернутый ответ на вопросы экзаменационного биле-

та. Преподаватель вправе задавать дополнительные вопросы по всем разделам изучаемой дисциплины. Во время ответа студент должен продемонстрировать знания по всему материалу изучаемой дисциплины. Студент должен уметь разделять факты и их интерпретацию, владеть методами аргументирования своих утверждений, а также методами построения и анализа математических моделей технологических процессов и производственных систем. Полнота ответа определяется показателями оценивания планируемых результатов обучения (раздел 1 «Фонда оценочных средств»).

При проведении промежуточной аттестации:

- ответ на «отлично» оценивается от 24 до 30 баллов;
- ответ на «хорошо» оценивается от 16 до 23 баллов;
- ответ на «удовлетворительно» оценивается от 8 до 15 баллов;
- ответ на «неудовлетворительно» оценивается от 0 до 7 баллов.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за 1 семестр по дисциплине «Принципы расширения стандартных прикладных программ» составляет 100 баллов.

Пересчет полученной студентом суммы баллов по дисциплине «Принципы расширения стандартных прикладных программ» в оценку (зачёт) осуществляется в соответствии с таблицей 2.

Таблица 2 — пересчета полученной студентом суммы баллов по дисциплине «Принципы расширения возможностей стандартных прикладных программ» в зачёт	
количество баллов	оценка
40–100	«зачтено»
0–39	«незачтено»

Текущие индивидуально набранные студентами баллы доводятся до их сведения 2 раза за семестр: в конце 6 и 12 недель обучения. Оценка студентам, успешно прошедшим обучение по дисциплине «Принципы расширения стандартных прикладных программ», может быть проставлена без сдачи зачёта на основании рейтинговой оценки по решению преподавателя.

7.2 Контрольные вопросы

1. Понятие алгоритма.

2. Переменные и их типы, правила составления идентификаторов.
3. Оператор присваивания.
4. Арифметические операторы.
5. Стандартные функции.
6. Соответствие типов, допустимые и недопустимые присваивания.
7. Структура программы.
8. Ввод/вывод в текстовом режиме.
9. Условный оператор. Операторы сравнения.
10. Логический тип данных и логические выражения. Логические операторы. Вычисление значения логических выражений.
11. Цикл со счётчиком, основные правила.
12. Операторы досрочного завершения цикла и итерации, цикл с уменьшающимся счётчиком.
13. Приём накопления суммы и произведения.
14. Вычисления с помощью рекуррентных соотношений.
15. Запись в текстовый файл чисел.
16. Чтение чисел из текстового файла.
17. Многомерные рекуррентные отношения.
18. Создание массива, выделение памяти.
19. Стандартные функции с массивами.
20. Чтение и запись из файла в массив.
21. Использование записей. Производные типы.
22. Символы, кодировки, вычисление символа по коду и кода по символу.
23. Строки, стандартные функции со строками. Явное преобразование типов.
24. Стандартные модули, библиотечные функции для работы с числовыми типами.

8 Учебно-методическое и информационное обеспечение дисциплины

8.1 Основная литература

1. Кудинов, Ю.И. Основы современной информатики. [Электронный ресурс] : Учебные пособия / Ю.И. Кудинов, Ф.Ф. Пащенко. — Электрон. дан. — СПб. : Лань, 2011. — 256 с. — Режим доступа: <http://e.lanbook.com/book/2024>
2. Дель, Л. А. Основы программирования. Ч. 1 : метод. указания к лаб. работам / Р. Н. Салимова, Оренбургский гос. ун-т, Л. А. Дель. — Оренбург : ОГУ, 2014. — Режим доступа: <http://rucont.ru/efd/271384>
3. Дель, Л. А. Основы программирования. Ч. 2 : метод. указания к лаб. работам / Р. Н. Салимова, Оренбургский гос. ун-т, Л. А. Дель. — Оренбург : ОГУ, 2014. — Режим доступа: <http://rucont.ru/efd/271385>

8.2 Дополнительная литература

1. Кудинов, Ю.И. Практикум по основам современной информатики. [Электронный ресурс] : Учебные пособия / Ю.И. Кудинов, Ф.Ф. Пащенко, А.Ю. Келина. — Электрон. дан. — СПб. : Лань, 2011. — 352 с. — Режим доступа: <http://e.lanbook.com/book/1799>
2. Васильчиков, В. В. Основы программирования на языке С : учеб. пособие / Н. С. Лагутина, Ю. А. Ларина, Яросл. гос. ун-т, В. В. Васильчиков. — Ярославль : ЯрГУ, 2006. — Режим доступа: <http://rucont.ru/efd/200112>.
3. Основы программирования в системе Pascal ABC / А.В. Копытин, Д.И. Соломатин, А.И. Другалев. — Воронеж : Издательско-полиграфический центр Воронежского государственного университета, 2009. — 79 с. — Режим доступа: <http://rucont.ru/efd/245181>

8.3 Программное обеспечение и Интернет-ресурсы

1. Операционная система семейства Windows, либо Linux, либо Mac OS X.
2. Интернет браузер, например, Firefox, Google Chrome/Chromium или иные совместимые.

3. На выбор компилятор языка программирования и среда разработки, распространяемые под свободными лицензиями (GPL, LGPL, BSD, MIT и совместимыми с ними):
 - (a) Free Pascal и среда разработки Lazarus;
 - (b) среда разработки Code::Blocks + компиляторы C, C++, D, Fortran;
 - (c) OpenJDK Java или Scala + среда Eclipse;
 - (d) Python + среда разработки IDLE.
4. Зональная научная библиотека им. В.А. Артисевич Саратовского государственного университета им. Н.Г. Чернышевского. – Режим доступа: <http://library.sgu.ru/>

9 Материально-техническое обеспечение дисциплины

Лекционные занятия по дисциплине «Принципы расширения стандартных прикладных программ» проводятся в аудиториях при помощи мультимедийного проектора, а также доски. Перед лекциями студентам может выдаваться печатный вариант краткого конспекта лекций.

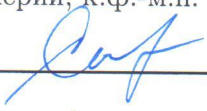
Лабораторные занятия со студентами проходят в компьютерном классе, где установлено необходимое программное обеспечение и имеется достаточное количество персональных компьютеров. Также в классе имеется маркерная доска для объяснения студентам рабочих общих моментов практикума.

Самостоятельная работа студента с теоретическим материалом возможна с использованием электронных ресурсов дома, в научной библиотеке, и других местах, постоянный доступ в Интернет при этом не обязателен. Также студент может дома решать задачи лабораторного практикума, используя домашний персональный компьютер, поскольку все задания практикума могут быть выполнены с использованием открытого свободного программного обеспечения.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению 11.03.04 «Электроника и нанoeлектроника» и профилям подготовки «Приборы микро- и нанoeлектроники, методы измерения микро- и наносистем», «Физика и технология твердотельных электронных микро- и наноструктур», «Микро- и нанoeлектроника, диагностика нано- и биомедицинских систем».

Программа одобрена на заседании кафедры динамического моделирования и биомедицинской инженерии 23 августа 2016 г., протокол № 8.

Автор — доцент кафедры динамического моделирования и биомедицинской инженерии, к.ф.-м.н. Сысоев И. В.



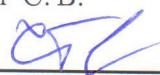
« ____ » _____ 2016 г.

Зав. кафедрой динамического моделирования и биомедицинской инженерии, д.ф.-м.н., доцент
Селезнёв Е. П.



« ____ » _____ 2016 г.

Декан факультета нано- и биомедицинских технологий, д.ф.-м.н., профессор Вениг С. Б.



« ____ » _____ 2016 г.