

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)



УТВЕРЖДАЮ

Проректор по ОД
А.А. Панфилов

« 03 » 09 2018 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«ЯЗЫКИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ»

Направление подготовки 02.03.02 Фундаментальная информатика и информационные техно-
логии

Профиль/программа подготовки

Уровень высшего образования бакалавриат

Форма обучения очная, ускоренная

Семестр	Трудоемкость зач. ед./ час.	Лекции, час.	Практич. занятия, час.	Лаборат. работы, час.	СРС, час.	Форма промежуточного контроля (экз./зачет)
1	2/72	-	-	-	72	Зачёт (переаттестация)
1	4/144	36	-	18	54	Экзамен (36 часов)
Итого	6/216	36	-	18	126	Зачёт (переаттестация), Экзамен (36 часов)

Владимир, 2018

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целями освоения дисциплины (модуля) «Языки и методы программирования» являются изучение студентами парадигм программирования, современных языков и методов программирования. Выработка у студентов умения самостоятельно реализовывать алгоритмы на языке программирования высокого уровня для решения прикладных задач, используя при этом наиболее подходящие методы программирования, уметь классифицировать данные и выбирать нужный класс данных для решения определенной задачи.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина входит в блок Б1 базовой части дисциплин ОПОП. Содержание дисциплины охватывает круг вопросов, связанных с разработкой и реализацией алгоритмов решения прикладных задач программирования, создания программных продуктов. Для успешного освоения курса студенты должны владеть знаниями и навыками, полученными ими при изучении школьного курса «Информатика и ИКТ», информационных дисциплин, изучаемых ими во время обучения в СПО, необходимо знание основных алгоритмических структур. Знания и умения, приобретенные студентами в результате изучения дисциплины, будут ими использоваться при изучении следующих курсов: «Объектно-ориентированное программирование», «Математическое моделирование», «Интеллектуальный анализ данных», написании выпускной квалификационной работы.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

способность к самоорганизации и самообразованию (ОК-7);

способность применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий (ОПК-2);

способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (ОПК-3);

способность использовать современные инструментальные и вычислительные средства (ПК-3);

способность решать задачи профессиональной деятельности в составе научно-исследовательского и производственного коллектива (ПК-4).

В результате освоения дисциплины студент должен демонстрировать освоение указанных компетенций по дескрипторам «знания, умения, владения», в соответствии с тематическими модулями дисциплины, применять полученные знания в последующем обучении и профессиональной деятельности:

- 1) Знать: содержание процессов самоорганизации и самообразования, их особенностей и технологий реализации, исходя из целей совершенствования профессиональной деятельности (ОК-7); основы современных языков программирования основные методы программирования, (ОПК-2); основные направления развития технологий программирования (ОПК-3); современные инструментальные и вычислительные средства (ПК-3); правила работы в составе научно-исследовательского и производственного коллектива; – метода и способы решения задач профессиональной деятельности (ПК-4).
- 2) Уметь: планировать цели и устанавливать приоритеты при выборе способов принятия решений с учетом условий, средств, личностных возможностей и временной перспективы достижения, осуществления деятельности; самостоятельно строить процесс овладения информацией, отобранной и структурированной для выполнения профессиональной деятельности (ОК-7); реализовывать алгоритмические решения на современном языке программирования (ОПК-2); анализировать проблемы и направления развития технологий программирования (ОПК-3); использовать в профессиональной деятельности современные инструментальные и вычислительные средства (ПК-3); работать в составе научно-исследовательского и производственного коллектива и решать задачи профессиональной деятельности (ПК-4).
- 3) Владеть: приемами саморегуляции эмоциональных и функциональных состояний при выполнении профессиональной деятельности; технологиями организации процесса самообразования; приемами целеполагания во временной перспективе, способами планирования, организации (ОК-7); базовыми средствами современных языков программирования, многомодульным программированием (ОПК-2); основными направлениями развития технологий программирования (ОПК-3); способностью использовать современные инструментальные и вычислительные средства (ПК-3); навыками работы в составе научно-исследовательского и производственного коллектива, алгоритмами решения задач профессиональной деятельности (ПК-4).

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины составляет 6 зачетных единиц, 216 часов.

№ п/п	Раздел (тема) дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Объем учебной работы, с применением интерактивных методов (в часах / %)	Формы текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	Контрольные работы	СРС	КП / КР		
1.	История развития языков программирования, обзор современных языков и методов программирования. Базовые средства языка C++	1	1-6	-	-	-	-	72	-	-	Переаттестация
2.	Структура программы на языке C++. Базовые типы данных. Операторы, приоритет операций.	1	1-5	10	-	4	-	10	-	7/50	
3.	Типы данных определяемые пользователем	1	6-11	12	-	4	-	10	-	4/25	Рейтинг-контроль №1
4.	Функции	1	12-13	4	-	2	-	6	-	2/33	Рейтинг-контроль №2
5.	Динамические структуры данных	1	14-15	4	-	4	-	11	-	2/25	
6.	Файлы	1	16	2	-	2	-	8	-	1/25	
7.	Введение в язык Python.	1	17-18	4	-	2	-	9	-	2/33	Рейтинг-контроль №3
Всего:		1	18	36	-	18	-	126	-	18/33	Зачёт (переаттестация), экзамен

Лекции

1. Введение. История развития языков программирования, обзор современных языков и методов программирования. (2 часа)
2. Структура программы на языке C++ (2 часа)
3. Классификация типов данных. Базовые типы данных. (2 часа)
4. Переменные и выражения. Операторы, приоритет операций. (2 часа)
5. Область действия и классы памяти. (2 часа)
6. Указатели. Ссылки. (4 часа)
7. Массивы. (4 часа)
8. Пользовательские типы данных. Перечисления. Структуры. Объединения. (4 часа)
9. Функции. Объявление и определение. Передача параметров. Возвращаемое значение. (2 часа)
10. Функции. Перегрузка функций. Шаблоны функций. (2 часа)

11. Динамические структуры данных. Линейный списки. Стек Очередь. Бинарные деревья (4 часа)
12. Работа с файлами. (2 часа)
13. Введение в язык Python. (4 часа)

Лабораторные работы.

1. Операторы. Приоритет операций (4 часа)
2. Указатели и ссылки. (2 часа)
3. Массивы данных. (2 часа)
4. Составные типы данных. Структуры. (2 часа)
5. Функции. (2 часа)
6. Динамические структуры данных. (2 часа)
7. Файлы. (2 часа)
8. Основы языка Python (2 часа)

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Для успешного освоения студентами данного курса необходимо использование активных и интерактивных методов обучения с уделением должного внимания имитационным моделям, позволяющих наиболее эффективно организовать процесс учебно-познавательной и исследовательской деятельности студентов, способствующему наиболее глубокому овладению компетенциями.

Сочетание традиционных образовательных технологий в форме лекции с интерактивными занятиями проведения мероприятий контроля текущих знаний студентов (задания рейтинг контроля, своевременная защита лабораторных работ), позволяют наиболее глубоко вовлечь студентов в процесс познавательной, научно-исследовательской деятельности, эффективно и равномерно распределить время самостоятельной работы студентов в течение семестра.

Во время занятий предусмотрено проведение диалога и дискуссий на лекциях по различным вопросам работы со сложными структурами данных и развития современных языков программирования, ряду других вопросов.

6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Зачет (перееаттестация):

Задание на перееаттестацию.

Представить отчёт о самостоятельной работе, в которой рассмотреть следующие вопросы:

Способы представления типов данных. Расчёт диапазона представления всех целых типов данных языка C++. Расчёт сопроводить подробными комментариями. Переполнение типов. Вопрос исследовать для знаковых и без знаковых типов.

Решить задачу. Решение представить в виде блок-схемы и программного кода, используя базовые средства языка C++.

Задачи на переаттестацию. (Вариант задачи получить у преподавателя.)

1. Дана целочисленная прямоугольная матрица. Определить: количество строк, не содержащих ни одного нулевого элемента; максимальное из чисел, встречающихся в заданной матрице более одного раза.
 2. Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.
 3. Характеристикой строки целочисленной матрицы назовём сумму её положительных чётных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.
 4. Дана целочисленная прямоугольная матрица. Определить: количество столбцов, содержащих хотя бы один нулевой элемент; номер строки, в которой находится самая длинная серия одинаковых элементов.
 5. Дана целочисленная квадратная матрица. Определить: произведение элементов только в тех строках, которые не содержат отрицательных; максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.
 6. Дана целочисленная квадратная матрица. Определить: сумму элементов в тех столбцах, которые не содержат отрицательных элементов; минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.
 7. Дана целочисленная прямоугольная матрица. Определить: сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент; номера строк и столбцов всех седловых точек матрицы. Матрица A имеет седловую точку, если является минимальным элементом в i -ой строке и максимальным в j -ом столбце.
 8. Для заданной матрицы размером 8 на 8 найти такие k , что k -я строка матрицы совпадает с k -ым столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.
 9. Характеристикой столбца целочисленной матрицы назовём сумму модулей его отрицательных нечётных элементов. Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик.
 10. Найти сумму элементов в тех столбцах матрицы $n \times n$, которые содержат хотя бы один отрицательный элемент.
 11. Соседями элемента A_{ij} в матрице назовём элементы A_{kl} с $i-1 \leq k \leq i+1$, $j-1 \leq l \leq j+1$, $(k, l) \neq (i, j)$. Операция сглаживания матрицы даёт новую матрицу того же размера, каждый элемент которой получается как среднее арифметическое имеющихся соседей соответствующего элемента исходной матрицы. Построить результат сглаживания заданной вещественной матрицы размером 10 на 10. В сглаженной матрице найти сумму модулей элементов, разложенных ниже главной диагонали.
 12. Элемент матрицы называется локальным минимумом, если он строго меньше всех имеющихся у него соседей. Подсчитать количество локальных минимумов заданной матрицы размером 10 на 10. Найти сумму модулей элементов, расположенных выше главной диагонали.
 13. Коэффициенты системы линейных уравнений заданы в виде прямоугольной матрицы. С помощью допустимых преобразований привести систему к треугольному виду.
 14. Найти количество строк, прямоугольной матрицы среднее арифметическое элементов которых меньше заданной величины.
 15. Уплотнить заданную матрицу, удалив из неё строки и столбцы, заполненные нулями.
 16. Найти номер первой из строк, содержащих хотя бы один положительный элемент.
- Вариант 13.
17. Осуществить циклический сдвиг элементов прямоугольной матрицы на n элементов вправо или вниз (в зависимости от введённого режима). n может быть больше количества элементов в строке или столбце.
- Вариант 14.

18. Осуществить циклический сдвиг элементов квадратной матрицы размерности $M \times N$ вправо на k элементов таким образом: элементы 1-й строки сдвигаются в последний столбец сверху вниз, из него – в последнюю строку; для остальных элементов – аналогично.
19. Дана целочисленная прямоугольная матрица. Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент.
20. Характеристикой строки целочисленной матрицы назовём сумму её отрицательных чётных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с убыванием характеристик.

Рейтинг-контроль №1

Вариант 1.

1. `int a=5; int b=a++; int c=++a;`
Каковы значения переменных a, b, c после выполнения этой строчки кода?
2. `int a=7; int b=27;`
`int k=(a>b)?printf("a+b=%i",a+b): printf("a-b=%i",a-b);`
Чему равно k ?
3. `signed char ch=129;`
`int a=ch++;`
Каковы значения переменных a и ch ?
4. `unsigned char i = 0, j = 256;`
`int k = j || i; int g = ~j;`
Каковы значения k и g ?

Вариант 2.

1. `int a=9; int b=a--; int c=--a;`
Чему равны значения переменных a, b, c после выполнения этой строчки кода?
2. `int a=37; int b=27;`
`int k=(a>b)?printf("a+b=%i",a+b++): printf("a-b=%i",a-b);`
Чему равно k ?
3. `unsigned char ch=255;`
`int a=ch++;`
Чему равны значения переменных a и ch ?
4. `signed char i =0, j = 374;`
`int k = j || i;`
`int g = ~j;`
Чему равны значения переменных k и g ?

Рейтинг-контроль №2

Вариант 1.

1. Написать фрагмент кода и схематически отобразить результат его работы. Объявить указатель на `int`, переменную типа `int`, настроить указатель на эту переменную и записать в нее через указатель число 10.
2. Объявить указатель на константный указатель на `int`, проинициализировать, сделать поясняющий рисунок.
3. Объявить одномерный массив `int` из пяти элементов, проинициализировать при объявлении. Сохранить в целочисленную переменную значение первого элемента массива. Показать два способа (индексное выражение и используя имя массива, как указатель)
4. Опишите синтаксис, работу и пример использования оператора цикла `while`

Вариант 2.

1. Написать фрагмент кода и схематически отобразить результат его работы. Объявить указатель на float, переменную типа float, настроить указатель на эту переменную и записать в нее через указатель число 10,5.
2. Объявить константный указатель на указатель на int, проинициализировать, сделать поясняющий рисунок.
3. Объявить одномерный массив int из пяти элементов, проинициализировать при объявлении. Сохранить в целочисленную переменную значение пятого элемента массива. Показать два способа (индексное выражение и используя имя массива, как указатель)
4. Опишите синтаксис, работу и пример использования оператора цикла do while

Рейтинг-контроль №3

Вариант 1

Организовать линейный список в виде стека с возможностью добавления и удаления элемента, вставки по ключу.

Вариант 2

Организовать линейный список в виде очереди с возможностью добавления и удаления элемента вставки по ключу.

Вопросы к экзамену.

1. Понятие тип данных. Базовые типы языка C++.
2. .Условный оператор. Синтаксис, пример использования.
3. Оператор выбора. Синтаксис, пример использования.
4. Цикл с предусловием. Синтаксис, пример использования.
5. Цикл с постусловием. Синтаксис, пример использования.
6. Цикл с параметром. Синтаксис, пример использования.
7. Массив. Определение, объявление, способы обращения к элементам.
8. Структуры. Определение, описание, объявление, способы обращения к полям.
9. Функции. Объявление, определение, вызов, формальные, фактические параметры.
10. Перегрузка функций.
11. Шаблоны функций.
12. Типы данных языка Python.
13. Локальные и глобальные переменные.
14. Организация циклов на Python.
15. Основные операции и операторы Python
16. Списки Python.
17. Массивы Python.
18. Печать данных по формату.
19. Работа с файлами на Python.
20. Функции в языке программирования Python. Параметры и аргументы функций.

Задания для самостоятельной работы студентов.

1. Найти периметр и площадь квадрата, факториал числа, интеграл (способы передачи параметров: по значению, по указателю, по ссылке, передача функции в качестве параметра).
2. Умножение матриц. Передача массивов в качестве параметров. 2 способа: статические и динамические массивы. Представление двумерного массива в виде вектора с использованием арифметики указателей.

3. Телефонный справочник. Массив структур. Реализовать функции: добавить, удалить, найти, вывести список. Использовать передачу параметров (массив и количество элементов, например).
4. Телефонный справочник. Связные списки. Реализовать функции: добавить элемент, добавить в указанное место, удалить заданный, найти, вывести список, обратить список. Использовать передачу параметров.
5. Найти периметр и площадь квадрата, факториал числа, интеграл (способы передачи параметров: глобальные переменные, по значению, по указателю, по ссылке, передача функции в качестве параметра). Объяснить в каждом случае, как передаются параметры, какой способ лучше для вычисления факториала.
6. Умножение матриц. Передача массивов в качестве параметров. 2 способа: статические и динамические массивы. Представление двумерного массива в виде вектора с использованием арифметики указателей.
7. Телефонный справочник. Использовать массив структур. Реализовать функции: добавить, удалить, найти, вывести список. Использовать передачу параметров (массив и количество элементов, например).
8. Телефонный справочник. Организовать с использованием связанного списка. Реализовать функции: добавить элемент, добавить в указанное место, удалить заданный, найти, вывести список, обратить список.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

а) основная литература:

1. Златопольский Д.М., Программирование: типовые задачи, алгоритмы, методы [Электронный ресурс] / Златопольский Д. М. - 3-е изд. (эл.). - М. : БИНОМ, 2015. - 226 с. - ISBN 978-5-9963-2932-8 - Режим доступа: <http://www.studentlibrary.ru/book/ISBN9785996329328.html>

2. С. М. Окулов Программирование в алгоритмах изд. (эл.). - М. : БИНОМ. Лаборатория знаний, -383 с. 2014 www.studentlibrary.ru

3. Давыдова Н.А. Программирование [Электронный ресурс]: учебное пособие/ Давыдова Н.А., Боровская Е.В.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, 2015.— 239 с.— Режим доступа: <http://www.iprbookshop.ru/6485>.— ЭБС «IPRbooks», 2015

б) дополнительная литература:

1. В.В. Подбельский Язык С#. Базовый курс: учеб. Пособие. М.: Финансы и статистика, - 408 с. 2015 www.studentlibrary.ru

2. В.В. Подбельский. Язык Си#. Решение задач: учеб. Пособие. - М.: Финансы и статистика. 296 с. 2014 <http://www.studentlibrary.ru>

3. Магда Ю.С. Программирование и отладка C/C++ приложений для микроконтроллеров ARM Издательство ДМК-пресс. 2012 <http://www.studentlibrary.ru/>

в) периодические издания:

1. «Информационные технологии» Ежемесячный теоретический и прикладной научно-технический журнал ISSN 1684-6400
2. Журнал «Вестник компьютерных и информационных технологий» ISSN 1810-7206
3. Журнал «Вестник ВлГУ» ISSN 2307-3241.

в) интернет-ресурсы:

1. Официальный сайт <https://www.python.org/>
2. Электронное пособие в открытом доступе. Столяров А.В. Введение в язык С++. МАКС-ПРЕСС 2011. <http://www.stolyarov.info>
3. Электронный ресурс, содержащий пошаговую инструкцию по установке и самоучитель Python <https://pythonworld.ru/osnovy/skachat-python.html>
4. Онлайн справочник программиста на С и С++ <http://www.c-cpp.ru/books>

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для успешного освоения дисциплины необходима аудитория для проведения лекционных занятий, имеющая необходимое количество посадочных мест и оснащенная доской для мела или маркера, компьютерный класс для проведения лабораторных работ, имеющих необходимое количество рабочих мест, с набором базового программного обеспечения для разработки программ на языке программирования С++.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению 02.03.02 Фундаментальная информатика и информационные технологии

Рабочую программу составил старший преподаватель каф. ФиПМ Шишкина М.В.

Рецензент

(представитель работодателя) Ген. директор ООО «ФС Сервис» Квасов Д.С.
(место работы, должность, ФИО, подпись)

Программа рассмотрена и одобрена на заседании кафедры ФиПМ

Протокол №1 от 03.09.2018 года

Заведующий кафедрой _____

С.М. Аракелян

(ФИО, подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии направления 02.03.02 Фундаментальная информатика и информационные технологии

Протокол №1 от 03.09.2018 года

Заведующий кафедрой _____

С.М. Аракелян

(ФИО, подпись)

ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ (МОДУЛЯ)

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____