

Федеральное государственное бюджетное образовательное учреждение
высшего образования
Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

**УТВЕРЖДАЮ**
Декан факультета ВМК МГУ
/И.А.Соколов/
_____ 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Наименование дисциплины (модуля):

Основы программирования

Уровень высшего образования:

бакалавриат

Направление подготовки (специальность):

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) ОПОП:

дисциплина относится к базовой части программы

Форма обучения:

очная

Москва 2023

Рабочая программа дисциплины (модуля) разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ бакалавриата, магистратуры, реализуемых последовательно по схеме интегрированной подготовки по направлениям 02.03.02, 02.04.02 «Фундаментальная информатика и информационные технологии» в редакции приказа МГУ от 30 декабря 2016 г.

1. Дисциплина относится к базовой части ОПОП ВО.

2. Входные требования для освоения дисциплины (модуля): отсутствуют.

3. Результаты обучения по дисциплине (модулю), соотношенные с требуемыми компетенциями выпускников.

Компетенции выпускников, частично формируемые при реализации дисциплины (модуля):

- **ОПК-2.** 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-3.** 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;
- **ПК-3.** Способность использовать современные инструментальные и вычислительные средства;
- **ПК-8.** Способность применять на практике международные и профессиональные стандарты информационных технологий, современные парадигмы и методологии, инструментальные и вычислительные средства.

Планируемые результаты обучения по дисциплине (модулю):

Знать:

1. язык программирования Си, его синтаксис, семантику, прагматику. Структуру программы на Си.
2. стандартную библиотеку языка Си.
3. базовые алгоритмы решения задач сортировки, поиска, работы с массивами, файлами.
4. основные структуры данных: стек, очередь, список, деревья и их применение в базовых алгоритмах.

Уметь:

1. разрабатывать и реализовывать многомодульные программы на языке Си, используя базовые алгоритмы, представленные на лекциях;
2. использовать возможности стандартной библиотеки языка Си;
3. обосновано выбирать алгоритмы решения задач и механизмы реализации конструкций языка Си;

Владеть:

1. современной технологией разработки и отладки многомодульных программ на языке Си.
4. Формат обучения: лекции проводятся с использованием компьютера и мультимедийного проектора для демонстрации примеров. ВО время семинаров и для самостоятельной работы студенты используют компьютеры в машинных залах и/или личные компьютеры.
5. Объем дисциплины (модуля) составляет 4 з.е., в том числе 72 академических часа, отведенных на контактную работу обучающихся с преподавателем, 72 академических часа на самостоятельную работу обучающихся.
6. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий.

Наименование и краткое содержание разделов и тем дисциплины (модуля), Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе				Самостоятельная работа обучающихся, часы (виды самостоятельной работы – эссе, реферат, контрольная работа и пр.)
		Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, часы			Всего	
		Занятия лекционного типа*	Занятия семинарского типа*	Всего		
1. Введение. Типы, константы, выражения языка Си. Операторы.	16	6	6	12	4	
2. Определение функций. Структура программы.	16	6	6	12	4	
3. Работа с указателями. Массивы. Строки.	18	6	6	12	6	
4. Схема трансляции программы. Препроцессор. Многомодульные программы	18	6	6	12	6	
5. Текущий контроль успеваемости: коллоквиум №1	2	0	0	0	2	
6. Определение новых типов в языке Си и их использование.	20	6	6	12	8	
7. Работа с файлами.	20	6	6	12	8	
8. Текущий контроль успеваемости: коллоквиум №2	2	0	0	0	2	
Промежуточная аттестация: устный экзамен	32	0	0	0	32	
Итого	144	36	36	72	44	

7. Фонд оценочных средств (ФОС) для оценивания результатов обучения по дисциплине (модулю)

7.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости.

Типовой вариант коллоквиума № 1

1. Полностью расставьте скобки в следующих выражениях (чтобы порядок вычислений был таким же, как и без скобок)

`a == b || a == c && c < 5`

`c = x != 0`

`a = -1 + + b -- - 5`

`a = b == c++`

`a[0] *= *b ? c : *d * 2`

`a - b, c = d`

2. Для каких значений `a` и `b` результаты выражений будут различны? Указать хотя бы одну пару таких `a` и `b`, и вычислить выражения при этих значениях `a` и `b`. Какие значения будут в `a` и `b` после вычисления каждого из выражений?

Выражение 1 : `a ++ && (b=a)`

Выражение 2 : `++a && (b=a)`

3. Улучшить стиль (структуру) фрагмента программы на Си (записать без оператора `continue`):

```
while ( E1 ) {  
    for (E2; E3; E4) {  
        S1  
        continue;  
        S2  
    }  
}
```

Замечание: здесь `E1`, `E2`, `E3`, `E4` - выражения допустимого в этом случае типа; `S1`, `S2` — операторы. не являющиеся операторами перехода и не содержащие внутри себя операторов перехода

4. Написать макрос `arith`, который для заданных целых или вещественных чисел вычисляет их среднее арифметическое.

Замечание: значение `arith(2, 4)` должно быть равно 3, значение `arith(1.2, 1.5)` должно быть равно 1.3.5.

5. Выписать результат работы препроцессора с файлом, содержащим следующую программу на Си.

```
#define K 1.0*  
#define M 1;
```

Ответ:

```

int main () {
    int i, s=0;
    for (i=M; i>0; i--) s*=i;
    #undef M
    { int M=-1;
      s*=M;
    }
    #ifndef M
    return 0;
    #else
    return 1;
    #endif
}

```

6. Определить, является ли правильной с точки зрения языка Си запись выражений вычеркнуть, для остальных вычислить их значения. (Переменные a, b описаны следующим образом: int a=0, b;) .

1 2 == (1, 3, 6)		(a+1) ++		(a=1) (a+2?:2:3)		b=0, b++, ++b

7. Описать функцию, вычисляющую значение $\max(x_0 + x_{n-1}, x_1 + x_{n-2}, x_2 + x_{n-3}, \dots, x_{(n-1)/2} + x_{n/2})$, где x_i - элементы вещественного массива x из n элементов.

8. Допустимо ли в Си? Если "да" - опишите семантику этих действий; если "нет" — объясните, почему.

```

void ques ( char * s1, char *s2, int n)
{ while (*s1 && *s2 && n-- && (*s1 ++ = *s2 ++ ) ); }

```

Типовой вариант коллоквиума № 2

1. Записать выражение, эквивалентное данному, без использования операции индексирования [].
p [2] [k]
2. Описать тип для представления списка целых чисел. Написать рекурсивную функцию, получающую в качестве параметра список и подсчитывающую количество четных чисел в этом списке.
3. Пусть «целочисленная» окружность на плоскости описана следующим образом:

```

struct point { int x; int y;};
struct circle { int radius; struct point center;};

```

Пусть есть массив `struct single P[50]`, содержащий информацию об окружностях на плоскости. Описать функцию, определяющую, есть ли среди этих окружностей хотя бы две концентрические (*имеющие один и тот же центр*) окружности.

4. Привести пример описания имен `x,y` и присвоить им начальные значения, чтобы выражение

`*x - &y == 2` было синтаксически корректным и результат его вычисления был равен 1.

5. Описать тип для работы с двоичными деревьями, в вершинах которых хранятся ключи – целые числа. Написать рекурсивную функцию, проверяющую, идентичны ли два заданных двоичных дерева. Деревья считать идентичными, если они имеют одинаковую структуру и одинаковые ключи в соответствующих вершинах.

7.2. Типовые контрольные задания или иные материалы для проведения промежуточной аттестации.

Список экзаменационных билетов.

Билет 1.

1. Представление логического типа в языке C. Логические и побитовые операции.
2. Функции: передача параметров и возврат значений.

Билет 2.

1. Понятие указателя в Си. Указатели и массивы (одномерные).
2. Этапы трансляции программы.

Билет 3.

1. Операторы цикла и перехода в языке Си.
2. Препроцессор языка Си, директивы условной компиляции. Защита файла от повторного включения.

Билет 4.

1. Работа с динамической памятью.
2. Классы памяти.

Билет 5.

1. Операции с побочным эффектом, точки последовательных вычислений.
2. Строки. Объявление, инициализация строки, перебор символов строки с использованием указателя.

Билет 6.

1. Представление в языке Си многомерных массивов и массивов указателей.
2. Преобразование типов при вычислениях арифметических выражений.

Билет 7.

1. Структуры. Операции, определенные для структур.
2. Квалификатор const

Билет 8.

1. Декларация typedef. Указатели на функции.
2. Ввод-вывод в программе на Си.

Билет 9.

1. Аргументы командной строки.
2. Препроцессор языка Си, директива #define .

Билет 10.

1. Структура программы на языке Си. Локализация переменных.
2. Операторы: условный, составной. Оператор выбора.

Типовые задачи, предлагаемые на устном экзамене

Задача 1.

Написать функцию, которая для заданного массива находит два наибольших элемента.

Задача 2.

Написать программу, распечатывающую последние символы аргументов командной строки.

Задача 3.

Написать функцию, которая для заданного массива целых чисел находит наибольший общий делитель этих чисел.

Задача 4.

Программа. Описать рекурсивную функцию вычисления $n!$ - факториала числа n , основанную на соотношении $n! = n*(n-1)!$. С ее помощью найти факториалы натуральных чисел от 1 до 5.

Задача 5.

Не используя функции стандартной библиотеки реализовать функцию, выполняющую сравнение строк типа const char* .

Задача 6.

Описать функцию, определяющую индекс первого элемента целочисленного массива из n элементов, значение которого равно заданному числу x . Если такого элемента в массиве нет, то считать номер равным -1 .

Задача 7.

Описать функцию, которая в заданной строке меняет местами ее первую и вторую половины.

Например, abcdefgh => efghabcd, vwxyz => yzxvww.

Задания, вынесенные на коллоквиумы и задачи итогового экзамена оцениваются единым образом. Максимальное количество баллов за задание – 6.

Набранные за решение заданий баллы суммируются, после чего определяется процент набранных баллов от максимального значения. Полученный процент определяет результирующую оценку.

0-39 %	40-59 %	60-79%	80-100 %
неудовлетворительно	удовлетворительно	хорошо	отлично

Итоговая оценка за курс определяется результатом устного экзамена.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине (модулю)				
Оценка	2	3	4	5
РО и соответствующие виды оценочных средств				
Знания <i>Коллоквиумы,</i> <i>Экзамен</i>	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения <i>Коллоквиумы,</i> <i>Экзамен</i>	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности непринципиального характера)	Успешное и систематическое умение
Навыки (владения, опыт деятельности) <i>Экзамен</i>	Отсутствие навыков (владений, опыта)	Наличие отдельных навыков (наличие фрагментарного опыта)	В целом, сформированные навыки (владения), но используемые не в активной форме	Сформированные навыки (владения), применяемые при решении задач

Соответствие результатов обучения и компетенций, в развитии которых участвует дисциплина (модуль)	
Результаты обучения	Компетенция, с частичным формированием которой связано достижение результата обучения
<p>Знать:</p> <ol style="list-style-type: none"> 1. Язык программирования С 2. Стандартную библиотеку языка Си <p>Уметь:</p> <ol style="list-style-type: none"> 1. Составлять и отлаживать многомодульные программы на языке Си. 	ОПК-2
<p>Знать:</p> <ol style="list-style-type: none"> 1. Особенности представления знаковых и беззнаковых чисел, явные и неявные преобразования типов; 2. Особенности реализации различных конструкций языка Си; 3. Особенности вычисления выражений <p>Уметь:</p> <ol style="list-style-type: none"> 1. Обосновано выбирать средства реализации алгоритма на языке Си и синтаксически и семантически корректно их записывать; 2. Оценивать эффективность работы алгоритмов. 	ОПК-3
<p>Владеть:</p> <p>Современной технологией разработки и отладки многомодульных программ на языке Си.</p>	ПК-3 ПК-8

8. Ресурсное обеспечение:

Основная литература:

1. Керниган Б., Ритчи Д. Язык программирования Си – изд-во «Диалектика», Санкт-Петербург, 2019 г.
2. Руденко Т.В. Сборник задач и упражнений по языку Си – Издательский отдел факультета ВМиК МГУ, Москва, 1999.
3. Робачевский А.М., Немногин С.А., Стесик О.Л. Операционная система UNIX, БХВ-Петербург, Санкт-Петербург, 2014.

Дополнительная литература:

1. Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. Алгоритмы. Построение и анализ. - «Вильямс», 2011
2. Д. Кнут. Искусство программирования для ЭВМ.
Т.1. Основные алгоритмы.
Т.2. Сортировка и поиск.- М., изд-во Вильямс, 2012

Материально-техническое обеспечение:

доска, компьютер и мультимедийный проектор для демонстрации слайдов в ходе лекций, компьютерный класс для проведения семинаров.

Программное обеспечение и Интернет-ресурсы:

задания семинарских занятий и вспомогательные материалы в системе moodle
moodle.cs.msu.ru — курс "Основы программирования"

9. Язык преподавания - русский.

10. Преподаватели:

доцент факультета ВМК МГУ Ю.С. Корухова

11. Авторы программ:

доцент факультета ВМК МГУ Ю.С. Корухова