

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Физический факультет

УТВЕРЖДАЮ:
Декан физического факультета
С.Н. Филимонов

Рабочая программа дисциплины

Языки программирования высокого уровня

по направлению подготовки

09.03.02 Информационные системы и технологии

Направленность (профиль) подготовки:

Цифровая астрономия и геоинформационные системы

Форма обучения

Очная

Квалификация

Инженер-разработчик информационных технологий

Год приема

2025

СОГЛАСОВАНО:
Руководитель ОП
О.М.Сюсина

Председатель УМК
О.М. Сюсина

Томск – 2025

1. Цель и планируемые результаты освоения дисциплины (модуля)

Целью освоения дисциплины является формирование следующих компетенций:

– ОПК-5 – Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.

Результатами обучения дисциплины являются::

– ООПК 5.1. – Знает алгоритмические языки программирования, операционные системы и платформы, современные среды разработки программного обеспечения;

– ООПК 5.2. – Умеет формулировать требования к прикладному программному обеспечению;

– ООПК 5.3. – Умеет разрабатывать алгоритмы и компьютерные программы для практического применения.

2. Задачи освоения дисциплины

– Освоить основные методы и алгоритмы программирования.

– Изучить синтаксические основы языка C++.

– Познакомиться с современным программным обеспечением, используемым при создании программного продукта.

– Приобрести практические умения и навыки по созданию собственных программных продуктов с использованием рассматриваемого программного обеспечения для решения практических задач профессиональной деятельности.

3. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина относится к обязательной части образовательной программы.

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Семестр 6, зачет.

Семестр 7, экзамен.

5. Входные требования для освоения дисциплины

Для успешного освоения дисциплины требуются компетенции, сформированные в ходе освоения образовательных программ предшествующего уровня образования.

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: Программирование, Технологии вычислительной физики, Практикум по вычислительной физике, Практикум по численным методам, Практикум по символьным вычислениям.

6. Язык реализации

Русский

7. Объем дисциплины (модуля)

Общая трудоемкость дисциплины составляет 7 з.е., 252 часа, из которых:

– лекции: 64 ч.;

– практические занятия: 64 ч.;

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины (модуля), структурированное по темам

Тема 1. Базовые средства и операторы языка C.

Базовые средства языка C: переменные, основные арифметические и логические операторы. Условный оператор. Операторы цикла. Функции в C.

Тема 2. Массивы

Одномерные числовые массивы. Массивы и указатели. Динамические массивы. Многомерные массивы фиксированного размера и динамические массивы. Строки. Массивы строк.

Тема 3. Работа с файлами средствами языка C

Схема работы с файлами, базовые функции, обеспечивающие работу с файлами.

Тема 4. Структуры. Список и стек.

Понятие структуры. Основные принципы работы со структурными типами данных. Список, как форма организации данных. Виды списков. Реализация списка с помощью структуры. Стек, как форма организации данных. Реализация стека на основе списка.

Тема 5. Основы ООП в C++

Основные принципы ООП. Понятие класса и объекта. Конструкторы и деструкторы. Скрытый указатель this.

Тема 6. Перегрузка операций

Перегрузка функций и операций. Основные принципы перегрузки операций. Дружественность. Исключительные ситуации и их обработка.

Тема 7. Классы-шаблоны. Агрегированные классы.

Функции-шаблоны и классы-шаблоны. Агрегированные классы: особенности конструкторов, особенности работы с член-данными.

Тема 8. Наследование в C++

Базовый и порожденные классы. Правила доступа в базовом и порожденном классе. Способы наследования. Ограничения наследования. Простое и множественное наследование. Виртуальный базовый класс. Стандартные преобразования при наследовании

Тема 9. Виртуальные функции

Раннее и позднее связывание. Виртуальные функции. Чистая виртуальная функция и абстрактный класс. Правила определения виртуальных функций. Механизм позднего связывания

Тема 10. Библиотека стандартных шаблонов

Структура БСШ. Класс vector. Класс list. Класс set. Класс queue.

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине проводится путем контроля посещаемости, проведения контрольных работ и тестов по лекционному материалу, выполнения домашних заданий и фиксируется в форме контрольной точки не менее одного раза в семестр.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Зачет в первом семестре проводится в письменной форме по билетам. Билет содержит теоретический вопрос и практическое задание. Продолжительность зачета 1,5 часа.

Примерный перечень теоретических вопросов

1. Базовые типы переменных в языке C, их объявление, инициализация. Неявное преобразование типов.
2. Основные арифметические и логические операции в языке C. Сокращенная форма записи арифметических операций. Операции инкремента и декремента, разница между их постфиксной и префиксной формой.
3. Форматированный ввод и вывод информации.
4. Условный оператор: краткая и полная форма, принцип работы.
5. Операторы цикла в C: цикл с предусловием, цикл с постусловием, параметрический цикл. Схема работы каждого оператора цикла. Условие работы цикла.
6. Функции в C: определение, системные и собственные функции. Как определяется функция. Вызов функции в программе. Упреждающее объявление функции.
7. Одномерный массив: определение, объявление массива фиксированного размера, нумерация элементов массива, размещение элементов массива в памяти. Инициализация элементов массива. Обращение к элементам массива. Массивы как аргументы функции.
8. Указатели: определение, объявление, операции для работы с указателем. Связь массивов и указателей. Одномерные динамические массивы: объявление, выделение памяти, расположение в памяти компьютера, инициализация, освобождение памяти. Понятие утечки памяти.
9. Понятие многомерного массива. Матрица, как пример двумерного массива: объявление матрицы фиксированного размера, нумерация элементов матрицы, размещение элементов матрицы в памяти. Инициализация элементов матрицы. Обращение к элементам матрицы. Матрицы фиксированного размера как аргументы функции.
10. Двумерные динамические массивы: объявление, выделение памяти, расположение в памяти компьютера, инициализация, освобождение памяти. Динамические матрицы как аргументы функции.
11. Тип `char`: назначение, инициализация переменных указанного типа. Кодировка символов и тип `char` как разновидность типа `int`.
12. Строка: определение, инициализация, объем занимаемой памяти, обращение к элементам строки. Базовые функции для работы со строками.
13. Файлы: схема работы с файлами, базовые функции, обеспечивающие работу с файлами.
14. Структуры: понятие структуры, основные принципы работы со структурными типами данных.
15. Список, как форма организации данных. Виды списков. Реализация списка с помощью структуры.
16. Стек, как форма организации данных. Реализация стека на основе списка.

Примеры практических заданий.

1. Даны 3 произвольных числа. Можно ли построить треугольник с такими длинами сторон, и, если можно, то какой (равносторонний, равнобедренный, обычный).
2. Найти K – количество различных делителей числа Z.
3. Вычислить
$$S = \sum_{i=1}^n \prod_{j=1}^m \frac{1}{i^2 + j^2}.$$
4. Выяснить, имеется ли в массиве C два идущих подряд нулевых элемента.
5. Найти в массиве самую длинную возрастающую последовательность, расположенную после максимального элемента. Вывести на экран номера ее первого и последнего элементов.
6. Найти сумму максимумов всех строк матрицы.
7. Удалить в матрице строки с нечетными номерами.

8. Дана строка $s_1, \dots, s_n$. Группы символов, разделенные пробелами (одним или несколькими) и не содержащие пробелов внутри себя, будем называть словами. Подсчитать количество слов в данной строке.

9. Дана строка $s_1, \dots, s_n$. Группы символов, разделенные пробелами (одним или несколькими) и не содержащие пробелов внутри себя, будем называть словами. Найти количество слов, у которых первый и последний символы совпадают между собой.

10. Создать файл из 20 случайных чисел разного знака. Выбрать из него во второй файл все положительные числа. Вывести оба файла на консоль.

11. Определить структуру, содержащую информацию о студенте:

```
struct STUD {char name[20]; int group, mark[5];};
```

Задать массив `STUD kurs[10]`, инициализировав его данными из файла (файл с данными `data.txt` создайте сами). Вывести список студентов, не имеющих аттестацию хотя бы по одному предмету.

12. Сформируйте однонаправленный список (с фиктивной головой) из n целых случайных чисел диапазона от 0 до 20 (число n вводится с клавиатуры). Добавьте в список заданный элемент после первого такого же элемента. В случае, если элемент в списке не присутствует, добавьте его в хвост.

Результаты зачета определяются оценками «зачтено» или «не зачтено».

Оценка «зачтено» выставляется, если:

а) ответ студента на теоретические вопросы полный или имеются незначительные замечания;

б) код практического задания верен или содержит ошибки синтаксического характера.

Оценка «не зачтено» выставляется, если:

а) ответ студента на теоретические вопросы не полный и содержит серьезные ошибки;

б) код практического задания содержит синтаксические и алгоритмические ошибки.

Если в течение семестра студент посетил не менее 75% занятий и выполнил все практические задания, то он освобождается от выполнения практической части билета.

Экзамен во втором семестре проводится в письменной форме по билетам. Экзаменационный билет состоит из двух частей. Продолжительность экзамена 1,5 часа.

Первая часть представляет собой два теоретических вопроса. Ответы на вопросы даются в развернутой форме и проверяют РОПК 5.1.

Вторая часть состоит из двух практических заданий и проверяет РОПК 5.2 и РОПК 5.3. Ответы на вопросы второй части предполагают написание программного кода для поставленной задачи и анализ его работы. Оценивается оптимальность выбранного для решения задачи алгоритма и скорость его работы.

Примерный перечень теоретических вопросов.

1. Принципы объектно-ориентированного программирования. Определение класса. Скрытие информации. Объект. Что за операция :: ? Чем член-функции отличаются от обычных? Какие существуют типы доступа и чем они отличаются друг от друга? Чем отличаются функции, определенные внутри класса, от функций, определенных вне класса?

2. Конструкторы и деструкторы. Назначение конструктора. Особенности конструктора. Конструктор копирования. Деструктор. Особенности деструктора. В каком случае необходим конструктор копирования и почему? Чем конструктор копирования отличается от перегрузки операции =?

3. Неявный указатель this. Перегрузка операций. Перечислите правила перегрузки операций. Перегрузки операций + и +=. Чем отличается операция + от операции +=?

4. Перегрузка операций. Перечислите правила перегрузки операций. Перегрузка операций [], ().

5. Перегрузка операций. Перечислите правила перегрузки операций. Перегрузка операции =. Чем конструктор копирования отличается от перегрузки операции =?

6. Дружественность. Что может быть другом класса? Перегрузка операций потокового ввода >> и вывода <<. Что общего и в чем разница между перегрузкой операции потокового вывода << и перегрузкой операции потокового ввода >>? Можно ли перегрузить операции потокового ввода/вывода как методы класса, ответ аргументировать. Обязательно ли использовать ссылку при перегрузке операций потокового ввода/вывода?

7. Массивы объектов. Какие конструкторы можно использовать и как? Можно ли явно инициализировать массив объектов, определенных в динамической памяти? Как работает конструктор для массива объектов? Как работает деструктор для массива объектов?

8. Функции-шаблоны и классы-шаблоны. Что такое порожденная функция? Использование функций-шаблонов и классов-шаблонов. Описать работу компилятора.

9. Агрегированные классы. Каким образом можно определить конструктор агрегированного класса, если член-данное в нем – указатель на объект другого (используемого) класса? Каким образом агрегированный класс может использовать член-данные используемого класса из части private?

10. Базовый и порожденный классы. Тип доступа protected. Типы наследования public и private. Ограничения наследования.

11. Простое наследование. Принцип доминирования в иерархии наследования.

12. Конструктор порожденного класса. Его вид. Стандартные преобразования при наследовании.

13. Множественное наследование. Прямые и не прямые базовые классы. Виртуальный базовый класс. Особенности инициализации его ч/данных.

14. Полиморфизм. Раннее и позднее связывание. Примеры.

15. Полиморфизм: виртуальные функции. Чистые виртуальные функции и абстрактный базовый класс

16. Правила определения виртуальных функций. Примеры.

Примеры практических заданий.

1. Написать следующие методы класса Array (Массив): конструктор без аргументов, конструкторы с аргументами, конструктор копирования, деструктор; перегрузка операторов << (вывод), = (присвоение.).

2. Написать следующие методы класса String: конструкторы, деструктор, перегрузка операций >> (поточный ввод), << (поточный вывод), = (присвоение), [], == (сравнение).

3. Написать следующие методы класса `Polinom`: конструкторы (все), перегрузка оператора присвоения, перегрузка операторов `*`(число) и `*=`(число).

4. Написать следующие методы класса `List` (список): конструктор по умолчанию, с аргументом (любой), конструктор копирования; перегрузка оператора `<<` (вывод).

5. Написать следующие методы класса `SET` (множество): конструктор копирования, перегрузка оператора - (удаление элемента), оператора `+=` (объединение).

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка «отлично» выставляется, если:

- а) студент дал полный и развернутый ответ на теоретические вопросы;
- б) код практического задания верен, оптимален (по скорости или по объему памяти), легко читаем, при написании кода использованы эффективные алгоритмы.

Оценка «хорошо» выставляется, если:

- а) ответ студента на теоретические вопросы в целом полный, но имеются незначительные замечания;

- б) код практического задания верен, но не оптимален (по скорости или по объему памяти), при написании кода использованы трудоемкие алгоритмы.

Оценка «удовлетворительно» выставляется, если:

- а) ответ студента на теоретические вопросы не полный;

- б) код практического задания содержит ошибки синтаксического характера.

Оценка «неудовлетворительно» выставляется, если:

- а) ответ студента на теоретические вопросы не полный и содержит серьезные ошибки;

- б) код практического задания содержит синтаксические и алгоритмические ошибки.

Если в течение семестра студент посетил не менее 75% занятий и выполнил все практические задания, то он освобождается от выполнения практической части билета.

11. Учебно-методическое обеспечение

- а) Электронный учебный курс по дисциплине в среде электронного обучения iDO

- б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

- в) План лекционных / практических занятий по дисциплине.

- г) Основная и дополнительная учебная литература.

12. Перечень учебной литературы и ресурсов сети Интернет

- а) основная литература:

– Рацеев С. М. Программирование на языке Си. / Рацеев С. М.. – Санкт-Петербург : Лань, 2022. – 332 с.. URL1: <https://e.lanbook.com/book/193320>.

– Огнева М. В. Программирование на языке C++: практический курс : Учебное пособие для вузов / Огнева М. В., Кудрина Е. В.. – Москва : Юрайт, 2022. – 335 с - (Высшее образование) . URL1: <https://urait.ru/bcode/492984>.

– Тузовский А. Ф. Объектно-ориентированное программирование : Учебное пособие для вузов / Тузовский А. Ф.. - Москва : Юрайт, 2022. – 206 с – (Высшее образование) . URL1: <https://urait.ru/bcode/490369>.

– Солдатенко И. С. Практическое введение в язык программирования Си / Солдатенко И. С., Попов И. В.. – Санкт-Петербург : Лань, 2021. – 132 с.. URL1: <https://e.lanbook.com/book/169287>.

- б) дополнительная литература:

– Кузин А. В. Программирование на языке Си : Справочник: Учебное пособие : Учебное пособие / Даичи ООО. - Москва : Издательство "ФОРУМ", 2020. - 143 с.. URL1: <http://znanium.com/catalog/document?id=362870>.

– Комлев Н. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей : Практическое пособие. - Москва : Издательство "СОЛОН-Пресс", 2020. - 298 с.. URL1: <http://znanium.com/catalog/document?id=392258>.

– Барков И. А. Объектно-ориентированное программирование : учебник / Барков И. А.. - Санкт-Петербург : Лань, 2019. - 700 с.. URL1: <https://e.lanbook.com/book/119661>.

– Баранова И. В. Объектно-ориентированное программирование на C++ : Учебник. - Красноярск : Сибирский федеральный университет, 2019. - 288 с.. URL1: <http://znanium.com/catalog/document?id=380554>.

– Ашарина И. В. Язык C++ и объектно-ориентированное программирование в C++ : лабораторный практикум : [учебное пособие для студентов вузов по направлению подготовки 09.03.01 - "Информатика и вычислительная техника"] / Ашарина И. В., Крупская Ж. Ф.. - Москва : Горячая линия - Телеком, 2016. - 231 с.: ил. - (Учебное пособие для высших учебных заведений. Специальность)

– Лафоре Р. Объектно-ориентированное программирование в C++ / Р. Лафоре. - 4-е изд.. - Санкт-Петербург [и др.] : Питер, 2016. - 923 с.: ил. - (Классика computer science)

– Павловская Т. А. C/C++. Процедурное и объектно-ориентированное программирование : [учебник для студентов вузов по направлению подготовки дипломированных специалистов "Информатика и вычислительная техника" : для бакалавров и специалистов] / Татьяна Павловская. - Санкт-Петербург [и др.] : Питер, 2015. - 495 с.: ил., табл. - (Стандарт третьего поколения) - (Учебник для вузов)

– Сибирякова В.А., Буторина Н.Б. Основы технологии объектно-ориентированного программирования на языке Си ++. Учебное пособие. Томск.: ТГУ, 2007. – 112 с.

в) ресурсы сети Интернет:

– Калинина Н., Костюкова Н. Основы программирования на языке C // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/43/43/info> (дата обращения: 21.02.2022)

– Александров Э., Афонин В. Программирование на языке C в Microsoft Visual Studio 2010 // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/627/483/info> (дата обращения: 21.02.2022)

– Брайен В., Деннис М. Язык программирования C // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/97/97/info> (дата обращения: 21.02.2022)

– Белоцерковская И., Галина Н., Катаева Л. Алгоритмизация. Введение в язык программирования C++ // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/16740/1301/info> (дата обращения: 21.02.2022)

– Фридман А. Язык программирования C++ // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/17/17/info> (дата обращения: 21.02.2022)

– Страуструп Б. Язык программирования C++ для профессионалов // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/98/98/info> (дата обращения: 21.02.2022)

– Павловская Т. Программирование на языке C++ // Национальный Открытый Университет «ИНТУИТ» – 2022. – URL: <https://intuit.ru/studies/courses/626/482/info> (дата обращения: 21.02.2022)

– Лесин В. Эффективное использование C++ // Просветительский проект «Лекториум» – 2022. – URL: <https://www.lektorium.tv/course/31228> (дата обращения: 21.02.2022)

- Линский Е. Основы С++. I семестр // Просветительский проект «Лекториум» – 2022. – URL: <https://www.lektorium.tv/course/22825> (дата обращения: 21.02.2022)
- Линский Е. Основы С++. II семестр // Просветительский проект «Лекториум» – 2022. – URL: <https://www.lektorium.tv/course/22858> (дата обращения: 21.02.2022)
- Общероссийская Сеть КонсультантПлюс Справочная правовая система. <http://www.consultant.ru>

13. Перечень информационных технологий

- а) лицензионное и свободно распространяемое программное обеспечение:
 - Microsoft Visual Studio 2017 (и выше)
 - Microsoft Office Standart 2013 Russian: пакет программ. Включает приложения: MS Office Word, MS Office Excel, MS Office PowerPoint, MS Office On-eNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook);
 - публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).
- б) информационные справочные системы:

– Электронный каталог Научной библиотеки	ТГУ	–
http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system		
– Электронная библиотека (репозиторий)	ТГУ	–
http://vital.lib.tsu.ru/vital/access/manager/Index		
– ЭБС Лань – http://e.lanbook.com/		
– ЭБС Консультант студента – http://www.studentlibrary.ru/		
– Образовательная платформа Юрайт – https://urait.ru/		
– ЭБС ZNANIUM.com – https://znanium.com/		
– ЭБС IPRbooks – http://www.iprbookshop.ru/		

14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

Занятия проводятся в компьютерном классе, оснащенном интерактивным оборудованием. Сеть Интернет используется для передачи информации между участниками учебного процесса и поиска необходимой информации.

15. Информация о разработчиках

Пахомова Елена Григорьевна, к.ф.-м.н., доцент, кафедра компьютерной безопасности, доцент