

**Федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»**

УТВЕРЖДАЮ
Директор ИКНК
_____ Д.П. Зегжда
«17» июня 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Технологии программирования»

Разработчик	Высшая школа программной инженерии
Направление (специальность) подготовки	09.03.04 Программная инженерия
Наименование ООП	09.03.04_01 Технология разработки и сопровождения качественного программного продукта
Квалификация (степень) выпускника	бакалавр
Образовательный стандарт	СУОС
Форма обучения	Очная

СОГЛАСОВАНО

Руководитель ОП

_____ А.В. Петров

«01» апреля 2025 г.

Соответствует СУОС

Утверждена протоколом заседания

высшей школы "ВШПИ"

от «01» апреля 2025 г. № 1

РПД разработал:

Старший преподаватель И.А. Шемякин

1. Цели и планируемые результаты изучения дисциплины

Цели освоения дисциплины

Основной целью курса «Технологии программирования» является формирование представления о принципах разработки качественного промышленного программного обеспечения с использованием объектно-ориентированного подхода на одном из языков высокого уровня (C++).

Результаты обучения выпускника

Код	Результат обучения (компетенция) выпускника ООП
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
ИД-8 ОПК-6	Использует средства стандартной библиотеки шаблонов для решения поставленных задач
ИД-9 ОПК-6	Разрабатывает программный код с использованием интегрированных сред разработки

Планируемые результаты изучения дисциплины

знания:

- Знает состав и принципы устройства основных компонентов стандартной библиотеки шаблонов
- Знает основные возможности современных интегрированных сред разработки

умения:

- Умеет применять элементы стандартной библиотеки шаблонов при разработке программного обеспечения
- Умеет создавать, конфигурировать и поддерживать проект с использованием современных интегрированных сред разработки

навыки:

- Владеет основными инструментами, входящими в состав современных интегрированных сред разработки

2. Место дисциплины в структуре ООП

В учебном плане дисциплина «Технологии программирования» относится к модулю «Модуль цифровых компетенций» / «Программирование».

Изучение дисциплины базируется на результатах освоения следующих дисциплин:

- Алгоритмизация и программирование

3. Распределение трудоёмкости освоения дисциплины по видам учебной работы и формы текущего контроля и промежуточной аттестации

3.1. Виды учебной работы

Виды учебной работы	Трудоёмкость по семестрам
	Очная форма
Лекционные занятия	14
Практические занятия	16
Самостоятельная работа	72
Промежуточная аттестация (зачет с оценкой)	6
Общая трудоёмкость освоения дисциплины	108, ач
	3, зет

3.2. Формы текущего контроля и промежуточной аттестации

Формы текущего контроля и промежуточной аттестации	Количество по семестрам
	Очная форма
Текущий контроль	
Расчетно-графические работы, шт.	1
Промежуточная аттестация	
Зачеты с оценкой, шт.	1

4. Содержание и результаты обучения

4.1 Разделы дисциплины и виды учебной работы

№ раздела	Разделы дисциплины, мероприятия текущего контроля	Очная форма		
		Лек, ач	Пр, ач	СР, ач
1.	Динамическая память и управление ресурсами	2	2	10
2.	Шаблоны и обобщенное программирование	2	2	10
3.	Потоки и строки std::string	2	2	10

4.	Итераторы стандартной библиотеки шаблонов (STL)	2	2	10
5.	Последовательные контейнеры стандартной библиотеки шаблонов (STL).	2	3	10
6.	Алгоритмы и функторы стандартной библиотеки шаблонов (STL)	2	3	12
7.	Ассоциативные контейнеры стандартной библиотеки шаблонов (STL)	2	2	10
Итого по видам учебной работы:		14	16	72
Зачеты с оценкой, ач				0
Часы на контроль, ач				0
Промежуточная аттестация (зачет с оценкой)		6		
Общая трудоёмкость освоения: ач / зет		108 / 3		

4.2. Содержание разделов и результаты изучения дисциплины

Раздел дисциплины	Содержание
1. Динамическая память и управление ресурсами	<p>Работа с динамической памятью в языках C и C++, особенности и отличия. Перегрузка функций new и delete, примеры. Копирование и перемещение объектов в C++, правило трёх/пяти. Гарантии безопасности исключений, идиома "копирование и обмен" (copy-and-swap). Идиома "умный указатель", создание собственного умного указателя, умные указатели стандартной библиотеки.</p> <p>Знание понятий и определений: динамическое выделение и освобождение памяти, управление ресурсами, умные указатели.</p> <p>Умение применять полученные знания при решении задач.</p>
2. Шаблоны и обобщенное программирование	<p>Необходимость шаблонов в C++. Инстанцирование. Шаблоны функций. Правила вывода типа. Идеальная передача (perfect forwarding). Шаблоны классов. Особые ситуации при работе с шаблонами: аргументы-не-типы (non-type), аргументы-шаблоны, зависимые имена типов. Шаблоны с переменным количеством аргументов. Концепты в C++20.</p> <p>Знание понятий и определений: обобщенное программирование, шаблоны функций, шаблоны классов, зависимые имена, специализации шаблонов, инстанцирование, особые аргументы шаблонов. Умение применять полученные знания при решении задач.</p>
3. Потоки и строки std::string	<p>Ввод/вывод в языке C и его проблемы. Функции библиотеки C для работы со строками. Класс std::string, его методы и внутреннее устройство. Поточковый ввод/вывод в языке C++. Флаги состояния потоков. Перегрузка операторов ввода/вывода. Идиома ScopeGuard Локализация</p> <p>Знание понятий и определений: концепция потока данных, устройство стандартного класса std::string. Умение применять полученные знания при решении задач.</p>
4. Итераторы стандартной библиотеки шаблонов (STL)	<p>Необходимость итераторов. Пример собственного итератора для связного списка. Категории итераторов стандартной библиотеки. Итераторы коллекций стандартной библиотеки. Обратные (реверсивные) итераторы. Итераторы ввода/вывода. Итераторы вставки. Инвалидация итераторов. Паттерн "итератор" (ООП).</p> <p>Знание понятий и определений: концепция итераторов, основные типы итераторов стандартной библиотеки шаблонов. Умение применять полученные знания при решении задач.</p>

5. Последовательные контейнеры стандартной библиотеки шаблонов (STL).	<p>Контейнеры стандартной библиотеки, требования.</p> <p>Последовательные контейнеры, требования. Устройство и особенности конкретных контейнеров. Аллокаторы. Инвалидация итераторов.</p> <p>Знание принципов работы и внутреннего устройства последовательных контейнеров стандартной библиотеки шаблонов. Умение применять полученные знания при решении задач.</p>
6. Алгоритмы и функторы стандартной библиотеки шаблонов (STL)	<p>Функции высших порядков и функции обратного вызова (callback). Функторы в C++. Привязка аргументов (binding). Замыкания. Лямбда-функции. Стандартные алгоритмы, примеры и особенности использования.</p> <p>Знание основных алгоритмов и функторов стандартной библиотеки шаблонов. Умение применять полученные знания при решении задач.</p>
7. Ассоциативные контейнеры стандартной библиотеки шаблонов (STL)	<p>Контейнеры стандартной библиотеки, требования. Ассоциативные контейнеры, требования. Устройство и особенности конкретных контейнеров. Неупорядоченные контейнеры, хеш-таблица. Инвалидация итераторов.</p> <p>Знание принципов работы и внутреннего устройства ассоциативных контейнеров стандартной библиотеки шаблонов. Умение применять полученные знания при решении задач.</p>

5. Образовательные технологии

В преподавании курса используются преимущественно традиционные образовательные технологии: лекции, лабораторные работы. Часть лабораторных работ реализуются в интерактивной форме с элементами индивидуального обучения и опережающей самостоятельной работы. Лекции и лабораторные работы могут проводиться как в очной, так и в дистанционной форме.

6. Лабораторный практикум

Не предусмотрено

7. Практические занятия

№ раздела	Наименование практических занятий (семинаров)	Трудоемкость, ач
		Очная форма
1.	Динамическая память и управление ресурсами.	2
2.	Шаблоны и обобщенное программирование.	2
3.	Потоки и строки <code>std::string</code> .	2
4.	Итераторы стандартной библиотеки шаблонов (STL).	2
5.	Последовательные контейнеры стандартной библиотеки шаблонов (STL).	2
6.	Алгоритмы и функторы стандартной библиотеки шаблонов (STL).	2
7.	Ассоциативные контейнеры стандартной библиотеки шаблонов (STL).	2
8.	Использование средств стандартной библиотеки шаблонов (STL).	2
Итого часов		16

8. Организация и учебно-методическое обеспечение самостоятельной работы

Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоемкость, ач
	Очная форма
Текущая СР	
работа с лекционным материалом, с учебной литературой	10
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	5
самостоятельное изучение разделов дисциплины	7
выполнение домашних заданий, домашних контрольных работ	32
подготовка к лабораторным работам, к практическим и семинарским занятиям	8
подготовка к контрольным работам, коллоквиумам	0
Итого текущей СР:	62
Творческая проблемно-ориентированная СР	
выполнение расчётно-графических работ	10
выполнение курсового проекта или курсовой работы	0
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
работа над междисциплинарным проектом	0
исследовательская работа, участие в конференциях, семинарах, олимпиадах	0
анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных	0
Итого творческой СР:	10
Общая трудоемкость СР:	72

9. Учебно-методическое обеспечение дисциплины

9.1. Адрес сайта курса

<https://dl.spbstu.ru/course/view.php?id=5981>

9.2. Рекомендуемая литература

Основная литература

№	Автор, название, место издания, издательство, год (годы) издания	Год изд.	Источник
1	Страуструп Б., Мартынов Н.Н. Язык программирования C++: Москва: Бином, 2015.	2015	ИБК СПбПУ
2	Шилдт Г., Ручко Н.М. C++. Базовый курс: Москва: Вильямс, 2015.	2015	ИБК СПбПУ

Ресурсы Интернета

1. ISO. ISO/IEC/IEEE 60559:2011 Information technology — Microprocessor Systems — Floating-Point arithmetic. Geneva, Switzerland: International Organization for Standardization, 2011, с. 58.: <https://www.iso.org/standard/57469.html>

9.3. Технические средства обеспечения дисциплины

Компилятор C++, поддерживающий стандарт C++14 (Рекомендуется Microsoft Visual Studio 2022)

Система контроля версий Git

10. Материально-техническое обеспечение дисциплины

Для успешного проведения лабораторных работ необходимо использование компьютерного класса, имеющего не менее 10 компьютеров, оснащенных необходимым программным обеспечением.

Для проведения занятий в дистанционном режиме каждому студенту необходима программа Microsoft Teams.

11. Критерии оценивания и оценочные средства

11.1. Критерии оценивания

Для дисциплины «Технологии программирования» формой аттестации является зачёт с оценкой. Дисциплина реализуется с применением системы индивидуальных достижений.

Текущий контроль успеваемости

Максимальное значение персонального суммарного результата обучения (ПСРО) по приведенной шкале - 100 баллов

Максимальное количество баллов приведенной шкалы по результатам прохождения двух точек контроля - 80 баллов.

Подробное описание правил проведения текущего контроля с указанием баллов по каждому контрольному мероприятию и критериев выставления оценки размещается в СДО в навигационном курсе дисциплины.

Промежуточная аттестация по дисциплине

Максимальное количество баллов по результатам проведения аттестационного испытания в период промежуточной аттестации – 20 баллов приведенной шкалы.

Промежуточная аттестация по дисциплине проводится в соответствии с расписанием.

Зачёт проводится в форме устного-письменного собеседования. Студенту задаются три вопроса, ответы на которые могут включать написание фрагментов кода. Примерное время ответа на билет 10-20 минут.

Критерии оценивания:

Оценка Требования	
5	На все вопросы даны полные ответы с примерами кода. Продемонстрировано глубокое понимание теоретического материала.
4	В ответах имеются незначительные неточности.
3	В ответах присутствуют неточности, примеры кода приведены с ошибками. Либо не дан правильный ответ на один из вопросов.
2	Не даны правильные ответы на два и более вопросов.

Результаты промежуточной аттестации, определяются на основе баллов, набранных в рамках применения, СИД

Баллы по приведенной шкале в рамках применения СИД (ПСРО+ ПА)	Оценка по результатам промежуточной аттестации
	Экзамен/диф.зачет/зачет
0 - 60 баллов	Неудовлетворительно/не зачтено
61 - 75 баллов	Удовлетворительно/зачтено
76 - 89 баллов	Хорошо/зачтено
90 и более	Отлично/зачтено

11.2. Оценочные средства

Оценочные средства по дисциплине представлены в фонде оценочных средств, который является неотъемлемой частью основной образовательной программы и размещается в электронной информационно-образовательной среде СПбПУ на портале etk.spbstu.ru

12. Методические рекомендации по организации изучения дисциплины

Успешное изучение курса требует посещения лекций, активной работы на лабораторных работах, выполнения всех учебных заданий преподавателя, ознакомления с основной и дополнительной литературой.

Во время лекции студент должен вести краткий конспект. Если какие-то материалы конспекта вызывают затруднения, необходимо постараться найти ответы на затруднительные вопросы, используя рекомендуемую литературу. Если студенту самостоятельно не удалось разобраться в материале, необходимо сформулировать вопросы и обратиться за помощью к преподавателю на консультации или ближайшей лекции. Обучающемуся необходимо регулярно отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам.

Лабораторные работы составляют важную часть профессиональной подготовки студентов. Они направлены на подтверждение теоретических положений и формирование учебных и профессиональных практических умений.

При подготовке к зачёту в дополнение к изучению конспектов лекций и учебных пособий, необходимо пользоваться учебной литературой, рекомендованной в настоящей программе. При подготовке к экзамену нужно изучить теорию: определения всех понятий и подходы к оцениванию до состояния понимания материала и самостоятельно решить по несколько типовых задач из каждой темы. При решении задач всегда необходимо уметь качественно интерпретировать итог решения.

13. Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.