**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Платформа Microsoft.NET  
Microsoft.NET platform

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 2

Регистрационный номер рабочей программы: 065068

Санкт-Петербург

2020

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Платформа Microsoft .NET занимает одно из ведущих мест в разработке программного обеспечения. В связи с этим курс представляет обучающимся комплекс знаний, умений и навыков, позволяющих овладеть процессом разработки современных приложений и освоить практическое применение стандартных инструментов, библиотек и средств инфраструктуры платформы Microsoft .NET.

Целями курса являются обучение обучающихся принципам разработки различных видов приложений на типо-безопасной (type safety) и управляемой (managed) платформе; формирование навыка использования современной среды разработки и стандартных библиотек; умение эффективно применять структуры данных и функциональные возможности языка C#; развитие способности дальнейшего освоения технологий, построенных на платформе Microsoft .NET.

Освоение программы курса необходимо для продолжения обучения другим дисциплинам в профиле, для успешной интеграции выпускников вузов в компании, занимающиеся промышленной разработкой программного обеспечения, а также для ведения полноценной исследовательской деятельности.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

Программа курса предназначена для обучающихся 4 курса бакалавриата и рассчитана на обучающихся, изучавших основы программирования, алгоритмов, структур данных и владеющих навыками создания программного обеспечения в рамках курсовых заданий.

Максимальная эффективность программы будет обеспечена при условии, что обучающийся:

* владеет навыками реализации программ (написания программного кода) на языках программирования C, С++, Java, C#.
* умеет применять базовые возможности современных сред разработки.

**1.3. Перечень результатов обучения (learning outcomes)**

После успешного окончания курса, обучающиеся будут обладать следующими навыками.

* Разрабатывать веб, настольные и мобильные приложения на платформе Microsoft.NET.
* Эффективно применять средства языка программирования C#, структуры данных стандартной библиотеки и среду разработки Microsoft Visual Studio.
* Проводить мониторинг и тестирование Microsoft.NET-приложений.
* Совместно работать в команде.

Дисциплина способствует развитию следующих компетенций:

* ОПК-1 – способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности;
* ОПК-3 – способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения;
* ОПК-4 – способен участвовать в разработке технической документации программных продуктов и программных комплексов;
* ОПК-5 – способен инсталлировать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства;
* ПКА-1 – способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий;
* ПКП-1 – способность проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности;
* ПКП-2 – способен решать задачи в области развития науки, техники и технологии с учетом нормативного правового регулирования в сфере интеллектуальной собственности;
* ПКП-3 – способен преподавать математику и информатику в средней школе, специальных учебных заведениях на основе полученного фундаментального образования и научного мировоззрения;
* ПКП-4 – способен применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях;
* ПКП-5 – способен использовать основные методы и средства автоматизации проектирования, реализации, испытаний и оценки качества при создании конкурентоспособного программного продукта и программных комплексов, а также способен использовать методы и средства автоматизации, связанные с сопровождением, администрированием и модернизацией программных продуктов и программных комплексов;
* ПКП-8 – способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования;
* УКБ-3 – способен понимать сущность и значение информации в развитии общества, использовать основные методы получения и работы с информацией с учетом современных технологий цифровой экономики и информационной безопасности.

**1.4. Перечень и объём активных и интерактивных форм учебных занятий**

Интерактивные лекции – 4 часа.

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | | | | Самостоятельная работа | | | | Объём активных и интерактивных  форм учебных занятий | Трудоёмкость |
| лекции | семинары | | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | | | | |
| Семестр 7 | 30 | |  |  |  |  |  |  |  | | 2 |  |  |  | 34 | |  | 6 |  | 4 | 2 |
|  | 2-100 | |  |  |  |  |  |  |  | | 2-100 |  |  |  | 1-1 | |  | 1-1 |  |  |  |
| ИТОГО | 30 | |  |  |  |  |  |  |  | | 2 |  |  |  | 34 | |  | 6 |  | 4 | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения очная | | | | | | |
| Семестр 7 |  |  | зачёт, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Наименование темы (раздела, части)** | **Вид учебных занятий** | **Кол-во часов** |
| I. | Введение в платформу Microsoft .NET | Лекции | 6 |
| Самостоятельная работа с использованием методических материалов | 7 |
| II. | Отличительные особенности языка программирования C# | Лекции | 6 |
| Самостоятельная работа с использованием методических материалов | 7 |
| III. | Программирование параллельных приложений на платформe Microsoft.NET | Лекции | 6 |
| Самостоятельная работа с использованием методических материалов | 7 |
| IV. | Работа с файловой системой | Лекции | 6 |
| Самостоятельная работа с использованием методических материалов | 7 |
| V. | Обзор инструментальных средств разработки для платформы Microsoft.NET | Лекции | 6 |
| Самостоятельная работа с использованием методических материалов | 6 |
| VI. | Промежуточная аттестация | Самостоятельная работа | 6 |
| Зачёт | 2 |
| **Итого** | | | **72** |

I. Введение в платформу Microsoft .NET

1. Цели, задачи и принципы .NET Framework, .NET Core.
2. Этапы развития платформы и языка C#.
3. Современное состояние платформы.
4. Применение для доступа к данным, разработки настольных (desktop), мобильных, веб и универсальных (UWP) приложений.

II. Отличительные особенности языка программирования C#

1. Основные принципы и основы.
2. Сравнительный анализ с языком C++.
3. Классы, структуры и интерфейсы.
4. Nullable-типы, частичные (partial) классы, методы расширения (extension methods).
5. Автоматическое управление памятью. Сборка мусора. Освобождение ресурсов: финализаторы и паттерн IDisposable.
6. Работа с неуправляемым (unmanaged) кодом: слабые ссылки (weak references), небезопасный (unsafe) код и использование Windows API в .NET-приложениях.
7. Обобщенное программирование (Generics).
8. Делегаты, лямбда-выражения и события.
9. Стандартные и специализированные коллекции.
10. Обработка данных с помощью LINQ.
11. Средства динамического программирования: пользовательские атрибуты, рефлексия (reflection), динамические (dynamic) типы.

III. Программирование параллельных приложений на платформe Microsoft.NET

1. Преимущества и проблемы многопоточных приложений.
2. Технологии низкоуровневого (Threads) и высокоуровневого параллелизма на уровне задач (Tasks). Класс Parallel. Технология PLINQ.
3. Основные средства синхронизации между потоками и процессами.
4. Принципы и примеры асинхронного программирования.

IV. Работа с файловой системой

1. Концепция и работа с потоками (streams).
2. Наблюдение за изменениями директорий и файлов через класс FileSystemWatcher.
3. Работа с файлами, отображаемыми в память (memory mapped files).
4. Коммуникация с помощью каналов (pipes).

V. Обзор инструментальных средств разработки для платформы Microsoft.NET

1. Программирование в среде разработки Microsoft Visual Studio.
2. Тестирование приложений с помощью Microsoft Visual Studio.
3. Диагностика и отладка приложений.
4. Применение IoC - контейнеров.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Для освоения дисциплины обучающиеся должны посещать лекции и практические занятия, выполнять задания преподавателей.

**3.1.2 Методическое обеспечение самостоятельной работы**

При самостоятельном изучении теоретического материала и выполнении практических заданий целесообразно использовать рекомендованную основную и дополнительную литературу.

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Промежуточная аттестация предполагает ответы обучающегося в устной форме на список экзаменационных вопросов, полученных в виде билета. Билет состоит из двух вопросов. Время подготовки ответа на вопросы билета составляет 45 минут.

Использование конспектов и учебников, а также электронных устройств хранения, обработки или передачи информации при подготовке и ответе на вопросы зачета категорически запрещено. В случае обнаружения факта использования недозволенных материалов (устройств) составляется акт и обучающийся удаляется с зачета. После ответа на вопросы билета преподаватель задает несколько дополнительных вопросов, на основании оценки ответов на которые итоговая оценка по предмету может быть повышена или понижена.

Критерии выставления оценок:

|  |  |  |
| --- | --- | --- |
| Количество правильных ответов на вопросы зачета | Оценка ECTS | Оценка при проведении зачёта |
| Больше 5 | A | зачтено |
| 4 | B |
| 3 | C |
| 2 | D |
| 1 | E |
| 0 | F | не зачтено |

* + 1. **Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

Список основных вопросов к зачету:

* + - 1. Цели, задачи и принципы .NET Framework, .NET Core.
      2. Этапы развития платформы и языка C#.
      3. Современное состояние платформы Microsoft.NET

1. Сравнительный анализ языков C# и C++.
2. Классы, структуры и интерфейсы.
3. Nullable-типы, частичные (partial) классы, методы расширения (extension methods).
4. Автоматическое управление памятью. Сборка мусора. Освобождение ресурсов: финализаторы и паттерн IDisposable.
5. Работа с неуправляемым (unmanaged) кодом: слабые ссылки (weak references), небезопасный (unsafe) код и использование Windows API в .NET-приложениях.
6. Обобщенное программирование (Generics).
7. Делегаты, лямбда-выражения и события.
8. Стандартные и специализированные коллекции.
9. Обработка данных с помощью LINQ.
10. Средства динамического программирования: пользовательские атрибуты, рефлексия (reflection), динамические (dynamic) типы.
11. Технологии низкоуровневого (Threads) и высокоуровневого параллелизма на уровне задач (Tasks). Класс Parallel. Технология PLINQ.
12. Основные средства синхронизации между потоками и процессами.
13. Принципы и примеры асинхронного программирования.
14. Концепция и работа с потоками (streams).
15. Наблюдение за изменениями директорий и файлов через класс FileSystemWatcher.
16. Работа с файлами, отображаемыми в память (memory mapped files).
17. Коммуникация с помощью каналов (pipes).
18. Применение IoC - контейнеров.

Список вопросов может варьироваться в зависимости от особенностей группы и фактически пройдённого материала.

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

Для оценки обучающимися содержания и качества учебного процесса применяется анкетирование в соответствии с методикой и графиком, утвержденными в установленном порядке.

**3.2. Кадровое обеспечение**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий**

К преподаванию дисциплины могут быть допущены преподаватели, имеющие диплом о высшем образовании по соответствующему направлению.

**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

Не требуется.

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий**

Требуется аудитория с мультимедийным оборудованием для демонстрации презентаций.

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

Доска для письма маркером, мультимедийный проектор.

**3.3.3 Характеристики специализированного оборудования**

Не требуется.

**3.3.4 Характеристики специализированного программного обеспечения**

Не требуется.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Маркеры для доски, губка.

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы**

1. Д.Рихтер. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. Изд-во: Питер, 2016. 896 с.
2. Э. Троелсен, Ф. Джепикс. Язык программирования C# 6.0 и платформа .NET 4.6. Изд-во: Вильямс, 2016. 1440 с.
3. Д. Албахари, Б. Албахари. C# 6.0. Справочник. Полное описание языка. Изд-во: Вильямс, 2016. 1040 с.

**3.4.2 Список дополнительной литературы**

Не требуется.

**3.4.3 Перечень иных информационных источников**

Ресурсы сети Интернет.

**Раздел 4. Разработчики программы**

Григорьев Дмитрий Алексеевич, доцент кафедры информатики, e-mail: [d.a.grigoriev](mailto:d.a.grigoriev)[@spbu.ru](mailto:n.golyandina@spbu.ru).