

**Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол от
30.11.2022 №13

Рабочая программа дисциплины

**Высокоуровневые методы
программирования**

(наименование дисциплины (модуля))

Уровень высшего образования
бакалавриат

(бакалавриат / магистратура / специалитет)

Направление подготовки / специальность

09.03.03 Прикладная информатика

(указывается код и наименование направления подготовки / специальности)

Направленность образовательной программы

Прикладная информатика в области принятия решений

(указывается профиль / магистерская программа / специализация)

Форма обучения

очная

(очная / очно-заочная / заочная)

Нижний Новгород

2022

1. Место дисциплины в структуре ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений

Б1.В.11 Высокоуровневые методы программирования

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции* (код, содержание индикатора)	Результаты обучения по дисциплине**	
ПК-4. Способен проводить исследование и описание процессов принятия решений в конкретной предметной (проблемной) области с применением современных информационных технологий, в том числе основанных на моделях и методах искусственного интеллекта	ПК-4.1. Демонстрирует знание современных моделей и методов интеллектуальной поддержки процессов принятия решений.	Знать базовые технологии, принципы и составляющие платформы .NET, синтаксис языка C#.	собеседование
	ПК-4.2. Демонстрирует умение применять системный подход к исследованию и описанию предметной (проблемной) области, формированию требований к ИС (ИИС) с учетом возможностей интеллектуальных технологий.	Уметь разрабатывать приложения под платформу .NET на языке программирования C# в среде MSVisualStudio	Собеседование задача
	ПК-4.3. Имеет практический опыт исследования и описания конкретной предметной области, разработки технического задания, эскизного и технического	Владеть представлениями (навыками) об основных технологиях .NET и их реализации на языке программирования C#	собеседование

	<i>проектов ИС (ИИС).</i>		
ПК-5. Способен проектировать интеллектуальные ИС (ИИС) по видам обеспечения	ПК-5.1. Демонстрирует знание современных технологий проектирования ИИС.	Знать методы программирования и внедрения прикладного программного обеспечения	собеседование
	ПК-5.2. Демонстрирует умение проектировать архитектуру ИИС по видам обеспечения.	Уметь программировать и внедрять прикладное программное обеспечение	Собеседование задача
	ПК-5.3. Имеет практический опыт проектирования конкретной ИИС по видам обеспечения.	Владеть навыками программированию и внедрению прикладного программного обеспечения	собеседование

3 Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная форма обучения
Общая трудоемкость	3 ЗЕТ
Часов по учебному плану	108
в том числе	
контактная работа:	65
- занятия лекционного типа	32
- занятия семинарского типа	32
- текущий контроль (КСР)	1
самостоятельная работа	43
Промежуточная аттестация – зачет	

3.2.Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины, форма промежуточной аттестации по дисциплине	Всего (часы)	в том числе	
		контактная работа (работа во взаимодействии с преподавателем), часы,из них	Самостоятельная работа студента часы

		Занятия лекционного типа	Занятия семинарского типа	Занятия лабораторного типа	Всего контактных часов	Всего СРС
Основные технологии Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Промежуточный язык (MS IL), Сборки (Assembly). Сборки из одного/нескольких файлов, Метаданные, Манифест, Компилятор времени выполнения (JIT), Выполнение приложения .NET, Структурные типы (value-based), Ссылочные типы (reference-based), Класс System.Object, Статические методы и методы экземпляров, Статические данные, Упаковка (boxing), Распаковка (unboxing), Пространства имен (namespace)	39	12		12	24	15
Коллекции Класс System.Array, Сортировка массивов, метод Array.Sort, Пользовательские нумераторы, Интерфейсы IEnumerable, IEnumerator, Пользовательский индексатор, Встроенные коллекции	35	10		10	20	15
Рефлексия Технология рефлексии типов, Класс System.Type, Получение объекта класса Type, Использование класса Type	33	10		10	20	13
Промежуточная аттестация - зачет	1				1	
Итого	108	32		32	65	43

Текущий контроль успеваемости реализуется в рамках лабораторных работ.
Промежуточная аттестация проходит в традиционной форме (зачет).

3. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа студента при изучении дисциплины «Высокоуровневые методы программирования (C#)» включает выполнение заданий под контролем преподавателя, решение домашних заданий и подготовку к экзамену.

Тематика самостоятельной работы

1. Реализация интерфейса ICloneable
2. Реализация интерфейса IComparable
3. Реализация интерфейса IComparer
4. Реализация интерфейсов IEnumerable, IEnumerator

Вопросы для самостоятельной работы

1. Интерфейс ICloneable
2. Интерфейс IComparable
3. Интерфейс IComparer
4. Класс System.Array
5. Сортировка массивов, метод Array.Sort
6. Пользовательские нумераторы. Интерфейсы IEnumerable, IEnumerator

4. Фонд оценочных средств для промежуточной аттестации по дисциплине, включающий:

4.1. Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатор а достижения компетенции)	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	Не зачтено		зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований . Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибки.	Уровень знаний в объеме, соответствующем программе подготовки . Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений . Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания но не в полном объеме.	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами .	Продемонстрированы все основные умения. Решены все основные задачи . Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продемонстрированы все основные умения, решены все основные задачи с отдельными несущественным недочетами, выполнены все задания в полном объеме.	Продемонстрированы все основные умения,. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов

<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продемонстрирован творческий подход к решению нестандартных задач
---------------	--	--	---	---	---	---	---

Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

5.2.1. Контрольные вопросы

Вопросы	Код формируемой компетенции
1. Common Language Runtime (CLR)	ПК-4
2. Common Type System (CTS)	ПК-4
3. Common Language Specification (CLS)	ПК-4
4. Промежуточный язык (MS IL)	ПК-4
5. Сборки (Assembly). Сборки из одного/нескольких файлов.	ПК-4
6. Метаданные	ПК-4
7. Манифест	ПК-4
8. Компилятор времени выполнения (JIT)	ПК-4
9. Выполнение приложения .NET	ПК-4
10. Структурные типы (value-based)	ПК-4
11. Ссылочные типы (reference-based)	ПК-4
12. Класс System.Object	ПК-4
13. Статические методы и методы экземпляров	ПК-4
14. Статические данные	ПК-4
15. Упаковка (boxing)	ПК-4
16. Распаковка (unboxing)	ПК-4
17. Пространства имен (namespace)	ПК-4
18. Область видимости на уровне типа	ПК-4
19. Область видимости на уровне члена	ПК-4
20. Свойства (properties)	ПК-4
21. Вложенные типы	ПК-4
22. Множественное наследование в C#	ПК-4
23. Запрет наследования класса	ПК-4
24. Перегрузка методов	ПК-4
25. Абстрактные классы и методы	ПК-4
26. Исключения, класс System.Exception	ПК-4
27. Генерация, перехват исключений.	ПК-4
28. Пользовательские исключения	ПК-4
29. Удаление объектов. Метод Finalize.	ПК-4
30. Интерфейс IDisposable	ПК-4

31. Сборщик мусора. Класс System.GC	ПК-5
32. Поколения объектов	ПК-5
33. Интерфейс.	ПК-5
34. Создание и реализация пользовательских интерфейсов	ПК-5
35. Получение ссылки на интерфейс	ПК-5
36. Обращение к членам интерфейса	ПК-5
37. Интерфейсы как параметры	ПК-5
38. Явная реализация интерфейсов	ПК-5
39. Интерфейс ICloneable	ПК-5
40. Интерфейс IComparable	ПК-5
41. Интерфейс IComparer	ПК-5
42. Класс System.Array	ПК-5
43. Сортировка массивов, метод Array.Sort	ПК-5
44. Пользовательские нумераторы. Интерфейсы IEnumerable, IEnumerator	ПК-5
45. Пользовательский индексатор	ПК-5
46. Встроенные коллекции	ПК-5
47. Создание делегатов	ПК-5
48. Класс System.MulticastDelegate	ПК-5
49. Создание многоадресных (комбинированных) делегатов	ПК-5
50. Использование делегатов	ПК-5
51. Применение событий	ПК-5
52. Создание событий	ПК-5
53. Организация срабатывания событий	ПК-5
54. Внутреннее представление событий	ПК-5
55. Прием событий	ПК-5
56. Объекты как приемники событий	ПК-5
57. Реализация обработки событий через интерфейсы	ПК-5
58. Технология рефлексии типов	ПК-5
59. Класс System.Type	ПК-5

60. Получение объекта класса Type	ПК-5
61. Использование класса Type	ПК-5

5.2.2. Типовые задания для оценки сформированности компетенции

Пример задания для оценивания компетенций ПК-4.

Знание интерфейса ICloneable

Реализация интерфейса ICloneable.

Пример задания для оценивания компетенций ПК-5.

Знание интерфейса IComparable

Реализация интерфейса IComparable.

Знание пользовательских индексаторов

Реализация интерфейсов IEnumerable, IEnumerator.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. **Марченко А. Л.** - Основы программирования на C# 2.0: учеб. пособие для студентов вузов, обучающихся по специальности "Приклад. информатика" и др. экон. специальностям. - М.: Интернет-Университет Информационных Технологий : Бином. Лаборатория знаний, (12 экз.)

7. Материально-техническое обеспечение дисциплины (модуля)

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ **09.03.03 «Прикладная информатика»**.

Автор _____ профессор Афраимович Л.Г.

Рецензент _____ профессор Федосенко Ю.С.

Заведующий кафедрой _____ профессор М.Х.Прилуцкий