



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
КГЭУ «КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КГЭУ»)

2 18.03.2025

«УТВЕРЖДАЮ»

И.о. директора института Теплоэнергетики
_____ Гапоненко С.О.

«11» октября 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Цифровое проектирование и решение инженерных задач на базе С#

(Наименование дисциплины в соответствии с РУП)

Направление подго-
товки

14.04.01 Ядерная энергетика и теплофизика
(Код и наименование направления подготовки)

Направленность (профиль)

Цифровой инжиниринг в атомной
энергетике
(Наименование направленности (профиля) образовательной программы)

Квалификация

Магистр

г. Казань, 2022

Рабочая программа дисциплины разработана в соответствии с ФГОС ВО по направлению подготовки 14.04.01 Ядерная энергетика и теплофизика (приказ Минобрнауки России от 27.03.2018 г. № 214).

Программу разработал(и):

Старший преподаватель, _____

канд.тех.наук

(должность, ученая степень)

(дата, подпись)

Сайтов С.Р.

(Фамилия И.О.)

Программа рассмотрена и одобрена на заседании выпускающей кафедры Атомные и тепловые электрические станции, протокол № 3-22/23 от 28.09.22

Заведующий кафедрой _____ Чичирова Н.Д.

(подпись)

Программа одобрена на заседании методического совета института Теплоэнергетики, протокол № 2 от 11.10.22.

Председатель методического совета
института Теплоэнергетики

_____ (подпись)

Гапоненко С.О.

Программа принята решением Ученого совета института Теплоэнергетики протокол № 2 от 11.10.22

1. Цель, задачи и планируемые результаты обучения по дисциплине

Целью освоения дисциплины является формирования навыков цифрового проектирования с упором на принципы объектно-ориентированного программирования (ООП).

Задачами дисциплины являются:

1. Получение навыков формулирования научно-технических задач в ядерной энергетике
2. Получение навыков решения научно-технических задач с помощью языков программирования на примере C#
3. Знакомство с концепцией ООП на примере C#.
4. Получение опыта разработки IT-проектов, в том числе на платформе .NET

Компетенции, формируемые у обучающихся, запланированные результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций:

| Код и наименование компетенции | Код и наименование индикатора достижения компетенции | Запланированные результаты обучения по дисциплине (знать, уметь, владеть) |
|---|--|--|
| Универсальные компетенции (УК) | | |
| УК-1 Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий | УК-1.2. Вырабатывает стратегию решения поставленной задачи (составляет модель, определяет ограничения, вырабатывает критерии, оценивает необходимость дополнительной информации) | <p><i>Знать:</i> Знает подходы к решению научно-технических задач (З1)</p> <p><i>Уметь:</i> Умеет определять ограничения, вырабатывать критерии и оценивать необходимость дополнительной информации в процессе решения поставленной задачи (У1)</p> <p><i>Владеть:</i> Владеет навыками разработки моделей исследуемых проблемных явлений (В1)</p> |
| | УК-1.3. Формирует возможные варианты решения задач | <p><i>Знать:</i> Знает методологию постановки задачи исследования (З1)</p> <p><i>Уметь:</i> Умеет применять современные информационные технологии, оргтехнику и средства связи для решения научно-технических задач (У1)</p> <p><i>Владеть:</i> Способен сформулировать возможные варианты решения научно-технической задачи (В1)</p> |

| Код и наименование компетенции | Код и наименование индикатора достижения компетенции | Запланированные результаты обучения по дисциплине (знать, уметь, владеть) |
|---|--|--|
| Общепрофессиональные компетенции (ОПК) | | |
| ОПК-2. Способен применять современные методы исследования, оценивать и представлять результаты выполненной работы | ОПК-2.4. Использует современные цифровые технологии для решения научно-технических задач | <i>Знать:</i> Знает принципы ООП (31) <i>Уметь:</i> Умеет применять современные информационные технологии (У1) <i>Владеть:</i> Владеет навыками цифрового проектирования на платформе .NET (В1) |

2. Место дисциплины в структуре ОПОП

Дисциплина «Цифровое проектирование и решение инженерных задач на базе C#» относится к обязательной части учебного плана по направлению подготовки 14.04.01 Цифровой инжиниринг в Атомной энергетике.

| Код компетенции | Предшествующие дисциплины (модули), практики, НИР, др. | Последующие дисциплины (модули), практики, НИР, др. ¹ |
|-----------------|--|--|
| УК-2 | | Управление IT-проектами |
| УК-3 | | Управление IT-проектами |
| УК-4 | | Управление IT-проектами |
| ПК-1 | | Инженерно-физическое моделирование технологических процессов АЭС |
| ПК-3 | | Инженерно-физическое моделирование технологических процессов АЭС |

Для освоения дисциплины обучающийся должен:

Знать: теоретические основы математического аппарата, основы алгоритмизации, технический иностранный (английский) язык

Уметь: пользоваться оргтехникой и средствами связи с сетью Интернет

Владеть: навыками процедурного программирования

3. Структура и содержание дисциплины

3.1. Структура дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных(ые) единиц(ы) (ЗЕ), всего 180 часа(ов), из которых 46 часа(ов) составляет контактная работа обучающегося с преподавателем (занятия лекционного типа 12 час., занятия семинарского типа (практические, семинарские занятия, лабораторные работы и т.п.) 34 часа, самостоятельная работа обучающегося 98 час.

¹ Перечисляются дисциплины (модули), практики, выполнение ВКР, др. по учебному плану, освоение которых базируется на результатах обучения по данной дисциплине.

| Вид учебной работы | Всего часов | Семестр(ы) |
|--|-------------|------------|
| | | 1 |
| ОБЩАЯ ТРУДОЕМКОСТЬ ДИСЦИПЛИНЫ | 180 | 180 |
| КОНТАКТНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ С ПРЕПОДАВАТЕЛЕМ, в том числе: | 46 | 46 |
| Лекции (Лек) | 12 | 12 |
| Практические (семинарские) занятия (Пр) | – | – |
| Лабораторные работы (Лаб) | 34 | 34 |
| САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ (СРС), в том числе: | 98 | 98 |
| Подготовка к промежуточной аттестации в форме: <i>экзамена</i> | 36 | 36 |
| ФОРМА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ (За – зачет, ЗО – зачет с оценкой, Э – экзамен) | Э | Э |

3.2. Содержание дисциплины, структурированное по разделам и видам занятий

| Разделы дисциплины | Семестр | Распределение трудоемкости (в часах) по видам учебной работы, включая СРС | | | | | | | | Формируемые результаты обучения (знания, умения, навыки) | Литература | Формы текущего контроля успеваемости | Формы промежуточной аттестации | Максимальное количество баллов по балльно - рейтинговой системе |
|--|---------|---|---|---------------------|------------------------|---|-------------------------|-------|----|--|------------|--------------------------------------|--------------------------------|---|
| | | Занятия лекционного типа | Занятия практического / семинарского типа | Лабораторные работы | Групповые консультации | Самостоятельная работа студента, в т.ч. подготовка к промежуточной аттестации | Сдача зачета / экзамена | Итого | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Раздел 1. Синтаксис С# | | | | | | | | | | | | | | |
| 1. Знакомство с платформой .NET | 1 | 2 | | 2 | | | | | 4 | УК-1.2-У1, В1, УК-1.3-У1, В1, ОПК 2.4- У1, В1 | Л.2.1 | ПЗ | Э | 12 |
| 2. Знакомство с синтаксисом С# | 1 | 2 | | 8 | | 25 | | | 35 | УК-1.2-31, УК-1.3-31 | Л.1.1, 2.1 | Тест | Э | |
| Раздел 2. Объектно-ориентированное программирование | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | |
|---|---|----|---|----|----|----|--|----|--|--------------|--------------|---|-----|
| 3. Основы ООП | 1 | 2 | | | | | | 2 | УК-1.2-31, УК-1.2-31, УК-1.3-У1, В1, Л.3-31, ОПК-31 | Л.1.1, Л.2.1 | Тест | Э | 24 |
| 4. Инкапсуляция | 1 | 2 | 8 | | 25 | | | 35 | УК-1.2-У1, В1, УК-1.3-У1, В1, ОПК 2.4- У1, В1 | Л.1.1, 2.1 | ПЗ, КЗ | Э | |
| Раздел 3. Наследование и полиморфизм в ООП | | | | | | | | | | | | | |
| 5. Наследование | 1 | 2 | 8 | | 25 | | | 35 | УК-1.2-У1, В1, УК-1.3-У1, В1, ОПК 2.4- У1, В1 | Л.1.1, Л.2.1 | ПЗ | Э | 24 |
| 6. Полиморфизм | 1 | 2 | 8 | | 23 | | | 33 | УК-1.2-31, У1, В1, УК-1.3-31, У1, В1, ОПК 2.4-31, У1, В1 | Л.1.1, Л.2.1 | Тест, ПЗ, КЗ | Э | |
| <i>Экзамен</i> | 1 | | | | | | | 36 | | | | | 40 |
| ИТОГО | | 12 | | 34 | | 98 | | 36 | | | | | 100 |

3.3. Тематический план лекционных занятий

| № п/п | Темы лекционных занятий | Трудоемкость, час. |
|--------------|------------------------------|--------------------|
| 1 | Знакомство с платформой .NET | 2 |
| 2 | Знакомство с синтаксисом C# | 2 |
| 3 | Основы ООП | 2 |
| 4 | Инкапсуляция | 2 |
| 5 | Наследование | 2 |
| 6 | Полиморфизм | 2 |
| Всего | | 12 |

3.4. Тематический план практических занятий

Данный вид работы не предусмотрен учебным планом.

3.5. Тематический план лабораторных работ

| № п/п | Темы лабораторных работ | Трудоемкость, час. |
|-------|-------------------------------|--------------------|
| 1 | Beginer.IRR | 2 |
| 2 | Incapsulation.Failures | 4 |
| 3 | Incapsulation.Photoshop | 4 |
| 4 | Incapsulation.RationalNumbers | 4 |
| 5 | Incapsulation.Weights | 4 |
| 6 | Inheritance.DataStucture | 4 |
| 7 | Inheritance.Geometry | 4 |

| | | |
|--------------|------------------------|-----------|
| 8 | Inheritance.MapObjects | 4 |
| 9 | Inheritance.Photoshop | 4 |
| Всего | | 34 |

3.6. Самостоятельная работа студента

| Номер раздела дисциплины | Вид СРС | Содержание СРС | Трудоемкость, час. |
|--------------------------|---|---|--------------------|
| 1 | Изучение синтаксиса С# | Передача параметров по значению и по ссылке, рекурсивные алгоритмы, сборки, манифесты, коллекции, обобщения | 25 |
| 2 | Разработка проекта с использованием принципа инкапсуляции | Пространства имен, классы, принципы декомпозиции, тестирование проекта | 25 |
| 3 | Разработка проекта с использованием принципа наследования | Интерфейсы, диаграммы классов | 25 |
| 3 | Разработка проекта с использованием принципа полиморфизма | Перегрузка методов, свойств, конструкторов | 23 |
| Всего | | | 98 |

4. Образовательные технологии

При проведении учебных занятий используются традиционные образовательные технологии (лекции в сочетании с лабораторными работами, самостоятельное изучение определённых разделов) и современные образовательные технологии, направленные на обеспечение развития у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств: интерактивные лекции, групповые дискуссии, анализ ситуаций и имитационных моделей, работа в команде, case-study, индивидуальное обучение, междисциплинарное обучение, опережающая самостоятельная работа.

5. Оценивание результатов обучения

Оценивание результатов обучения по дисциплине осуществляется в рамках текущего контроля успеваемости, проводимого по балльно-рейтинговой системе (БРС), и промежуточной аттестации.

Текущий контроль успеваемости осуществляется в течение семестра, включает: защиты лабораторных работ; защиты заданий, выполненных индивидуально или группой обучающихся; контроль самостоятельной работы обучающихся в устной форме.

Итоговой оценкой результатов освоения дисциплины является оценка, выставленная во время промежуточной аттестации обучающегося (экзамен) с учетом результатов текущего контроля успеваемости.

Промежуточная аттестация в форме экзамена проводится в виде тестирования за компьютером. На экзамен выносятся теоретические (тесты) и практические задания, проработанные в течение семестра на учебных занятиях и в процессе самостоятельной работы обучающихся. Процесс тестирования включает 10 теоретических вопросов и 1 практическое задание.

Обобщенные критерии и шкала оценивания уровня сформированности компетенции (индикатора достижения компетенции) по итогам освоения дисциплины:

| Планируемые результаты обучения | Обобщенные критерии и шкала оценивания результатов обучения | | | |
|---|--|---|---|---|
| | неудовлетворительно | удовлетворительно | хорошо | отлично |
| | не зачтено | зачтено | | |
| Полнота знаний | <i>Уровень знаний ниже минимальных требований, имеют место грубые ошибки</i> | <i>Минимально допустимый уровень знаний, имеет место много негрубых ошибок</i> | <i>Уровень знаний в объеме, соответствующем программе, имеет место несколько негрубых ошибок</i> | <i>Уровень знаний в объеме, соответствующем программе подготовки, без ошибок</i> |
| Наличие умений | <i>При решении стандартных задач не продемонстрированы основные умения, имеют место грубые ошибки</i> | <i>Продемонстрированы основные умения, решены типовые задачи с негрубыми ошибками, выполнены все задания, но не в полном объеме</i> | <i>Продемонстрированы все основные умения, решены все основные задачи с негрубыми ошибками, выполнены все задания в полном объеме, но некоторые с недочетами</i> | <i>Продемонстрированы все основные умения, решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме</i> |
| Наличие навыков (владение опытом) | <i>При решении стандартных задач не продемонстрированы базовые навыки, имеют место грубые ошибки</i> | <i>Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами</i> | <i>Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами</i> | <i>Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов</i> |
| Характеристика сформированности компетенции (индикатора достижения компетенции) | <i>Компетенция в полной мере не сформирована. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач</i> | <i>Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач, но требуется дополнительная практика по большинству практических задач</i> | <i>Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач</i> | <i>Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач</i> |

| | | | | |
|--|--------|---------------|---------|---------|
| Уровень сформированности компетенции (индикатора достижения компетенции) | Низкий | Ниже среднего | Средний | Высокий |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Шкала оценки результатов обучения по дисциплине:

| Код компетенции | Код индикатора достижения компетенции | Запланированные результаты обучения по дисциплине | Уровень сформированности компетенции (индикатора достижения компетенции) | | | |
|-----------------|---------------------------------------|---|--|---|--|---|
| | | | Высокий | Средний | Ниже среднего | Низкий |
| | | | Шкала оценивания | | | |
| | | | отлично | хорошо | удовлетворительно | неудовлетворительно |
| | | | зачтено | | | не зачтено |
| УК-1 | УК-1.2 | знать: | | | | |
| | | Знает подходы к решению научно-технических задач (31) | Знает подходы, применяет их на практике | Знает подходы, прибегает к чужой помощи при использовании их на практике | Знает подходы, но не способен применить их на практике | Не знает подходов к решению научно-технических задач |
| | | уметь: | | | | |
| | | Умеет определять ограничения, вырабатывать критерии и оценивать необходимость дополнительной информации в процессе решения поставленной задачи (У1) | Определяет ограничения, вырабатывает критерии и оценивает необходимость дополнительной информации в процессе решения поставленной задачи | Определяет ограничения, вырабатывает критерии решения поставленной задачи | Определяет критерии решения поставленной задачи | Не умеет определять ограничения, вырабатывать критерии и оценивать необходимость дополнительной информации в процессе решения |
| УК-1.3 | УК-1.3 | владеть: | | | | |
| | | Владеет навыками разработки моделей исследуемых проблемных явлений (В1) | Способен разработать достоверную модель исследуемого явления | Способен разработать модель исследуемого явления без верификации её достоверности | Разрабатывает модели явлений с серьезными неточностями | Не владеет навыками разработки моделей исследуемых проблемных явлений |
| | | знать: | | | | |
| | | Знает методологию постановки задачи | Знает методологию по- | Знает методологию, но | Знает методоло- | Не знает методологию |

| | | | | | | |
|-------|---------|--|---|--|--|--|
| | | исследования (31) | становки задачи и успешно применяет её на практике | с трудом применяет её на практике | гию, но не способен применить её на практике | постановки задачи |
| | | уметь: | | | | |
| | | Умеет применять современные информационные технологии, оргтехнику и средства связи для решения научно-технических задач (У1) | Успешно применяет современные ИТ технологии, оргтехнику и средства связи для решения научно-технических задач | Применяет ИТ-технологии, оргтехнику, но испытывает сложности при работе с сетевыми устройствами и интернетом | Испытывает сложности при работе с оргтехникой, низкая продуктивность при работе с ИТ-технологиями | Не умеет применять современные информационные технологии, оргтехнику и средства связи |
| | | владеть: | | | | |
| | | Способен сформулировать возможные варианты решения научно-технической задачи (В1) | Способен сформулировать ряд возможных решений и выбрать наиболее оптимальный | Способен сформулировать ряд возможных решений, но не способен выбрать оптимальный | Ограничивается единственным и не всегда оптимальным вариантом решения задачи | Не способен сформулировать вариант решения научно-технической задачи |
| ОПК-2 | ОПК-2.4 | знать: | | | | |
| | | Знает принципы ООП (31) | Знает принципы ООП, свободно применяет их на практике | Знает принципы ООП, прибегает к чужой помощи при их использовании | Знает принципы ООП, но не способен их применить | Не знает принципов ООП |
| | | уметь: | | | | |
| | | Умеет применять современные информационные технологии (У1) | Успешно и самостоятельно применяет современные информационные технологии для решения научно-технических задач | Применяет современные информационные технологии консультируясь с коллегами | Испытывает сложности при работе с информационными технологиями, требуется помощь со стороны коллег | Не способен применять современные информационные технологии для решения научно-технических задач |
| | | владеть: | | | | |
| | | Владеет навыками цифрового проектирования на платформе .NET (В1) | Полная работоспособность решения в соответствии с ТЗ | Частичная работоспособность решения, соответствующего ТЗ | Некорректная архитектура, либо отклонение от ТЗ, но способность про- | Не владеет навыками цифрового проектирования на платформе .NET |

| | | | | | | |
|--|--|--|--|--|-----------------------|--|
| | | | | | граммы скопировать | |
|--|--|--|--|--|-----------------------|--|

Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации приведены в Приложении к рабочей программе дисциплины. Полный комплект заданий и материалов, необходимых для оценивания результатов обучения по дисциплине, хранится на кафедре-разработчике в бумажном и электронном виде.

6. Учебно-методическое и информационное обеспечение дисциплины

6.1. Учебно-методическое обеспечение

Основная литература

| № п/п | Автор(ы) | Наименование | Вид издания (учебник, учебное пособие, др.) | Место издания, издательство | Год издания | Адрес электронного ресурса | Кол-во экземпляров в библиотеке КГЭУ |
|-------|---------------|--|---|-----------------------------|-------------|---|--------------------------------------|
| 1 | Залогова Л.А. | Основы объектно-ориентированного программирования на базе языка C# | Учебное пособие | СПб, М, Краснодар: "Лань" | 2021 | https://e.lanbook.com/book/176894 | |

Дополнительная литература

| № п/п | Автор(ы) | Наименование | Вид издания (учебник, учебное пособие, др.) | Место издания, издательство | Год издания | Адрес электронного ресурса | Кол-во экземпляров в библиотеке КГЭУ |
|-------|------------|----------------------------|---|-----------------------------|-------------|---|--------------------------------------|
| 1 | Вафин Р.Р. | Программирование на C#.NET | Учебное пособие | Казань : КНИТУ-КАИ | 2020 | https://e.lanbook.com/book/264917 | |

6.2. Информационное обеспечение

6.2.1. Электронные и интернет-ресурсы

| № п/п | Наименование электронных и интернет-ресурсов | Ссылка |
|-------|---|---|
| 1 | Электронный учебный курс Moodle2: Цифровое проектирование и решение инженерных задач на базе C# | https://lms2.kgeu.ru/course/view.php?id=211 |

6.2.2. Профессиональные базы данных

| № п/п | Наименование профессиональных баз данных | Адрес | Режим доступа |
|-------|--|-------|---------------|
|-------|--|-------|---------------|

| | | | |
|---|----------------------|---|----------|
| 1 | Документация по .NET | https://learn.microsoft.com/ru-ru/dotnet/ | Открытый |
| 2 | Документация по C# | https://learn.microsoft.com/ru-ru/dotnet/csharp/ | Открытый |

6.2.3. Информационно-справочные системы

| № п/п | Наименование информационно-справочных систем | Адрес | Режим доступа |
|-------|--|---|---------------|
| 1 | Перечень ошибок компилятора C# | https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/compiler-messages/ | Открытый |

6.2.4. Лицензионное и свободно распространяемое программное обеспечение дисциплины

| № п/п | Наименование программного обеспечения | Способ распространения (лицензионное/свободно) | Реквизиты подтверждающих документов |
|-------|--|--|---|
| 1 | Microsoft Visual Studio Community 2022, версия 17.13.2 | Свободно | https://visualstudio.microsoft.com/ru/vs/community/ |
| 2 | Microsoft .NET Framework, версия 4.8.04084 | Свободно | https://go.microsoft.com/fwlink/?linkid=2088631 |

7. Материально-техническое обеспечение дисциплины

| № п/п | Вид учебной работы | Наименование специальных помещений и помещений для СРС | Оснащенность специальных помещений и помещений для СРС |
|-------|-------------------------------------|--|--|
| 1 | Лекционные занятия | Учебная аудитория для проведения занятий лекционного типа с проектором и экраном | Учебная аудитория для проведения занятий лекционного типа с проектором и экраном |
| 2 | Лабораторные работы | Компьютерный класс с выходом в Интернет, проектором или телевизором | Компьютерный класс с выходом в Интернет, проектором или телевизором |
| 3 | Самостоятельная работа обучающегося | Компьютерный класс с выходом в Интернет, проектором или телевизором | Компьютерный класс с выходом в Интернет, проектором или телевизором |

8. Особенности организации образовательной деятельности для лиц с ограниченными возможностями здоровья и инвалидов

Лица с ограниченными возможностями здоровья (ОВЗ) и инвалиды имеют возможность беспрепятственно перемещаться из одного учебно-лабораторного корпуса в другой, подняться на все этажи учебно-лабораторных корпусов, заниматься в учебных и иных помещениях с учетом особенностей психофизического развития и состояния здоровья.

Для обучения лиц с ОВЗ и инвалидов, имеющих нарушения опорно-двигательного аппарата, обеспечены условия беспрепятственного доступа во все учебные помещения. Информация о специальных условиях, созданных для обучающихся с ОВЗ и инвалидов, размещена на сайте университета www/kgeu.ru. Имеется возможность оказания технической помощи ассистен-

том, а также услуг сурдопереводчиков и тифлосурдопереводчиков.

Для адаптации к восприятию лицами с ОВЗ и инвалидами с нарушенным слухом справочного, учебного материала по дисциплине обеспечиваются следующие условия:

- для лучшей ориентации в аудитории, применяются сигналы оповещения о начале и конце занятия (слово «звонок» пишется на доске);
- внимание слабослышащего обучающегося привлекается педагогом жестом (на плечо кладется рука, осуществляется нерезкое похлопывание);
- разговаривая с обучающимся, педагогический работник смотрит на него, говорит ясно, короткими предложениями, обеспечивая возможность чтения по губам.

Компенсация затруднений речевого и интеллектуального развития слабослышащих обучающихся проводится путем:

- использования схем, диаграмм, рисунков, компьютерных презентаций с гиперссылками, комментирующими отдельные компоненты изображения;
- регулярного применения упражнений на графическое выделение существенных признаков предметов и явлений;
- обеспечения возможности для обучающегося получить адресную консультацию по электронной почте по мере необходимости.

Для адаптации к восприятию лицами с ОВЗ и инвалидами с нарушениями зрения справочного, учебного, просветительского материала, предусмотренного образовательной программой по выбранному направлению подготовки, обеспечиваются следующие условия:

- ведется адаптация официального сайта в сети Интернет с учетом особых потребностей инвалидов по зрению, обеспечивается наличие крупношрифтовой справочной информации о расписании учебных занятий;
- педагогический работник, его собеседник (при необходимости), присутствующие на занятии, представляются обучающимся, при этом каждый раз называется тот, к кому педагогический работник обращается;
- действия, жесты, перемещения педагогического работника коротко и ясно комментируются;
- печатная информация предоставляется крупным шрифтом (от 18 пунктов), тотально озвучивается;
- обеспечивается необходимый уровень освещенности помещений;
- предоставляется возможность использовать компьютеры во время занятий и право записи объяснений на диктофон (по желанию обучающихся).

Форма проведения текущей и промежуточной аттестации для обучающихся с ОВЗ и инвалидов определяется педагогическим работником в соответствии с учебным планом. При необходимости обучающемуся с ОВЗ, инвалиду с учетом их индивидуальных психофизических особенностей дается возможность пройти промежуточную аттестацию устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п., либо предоставляется дополнительное время для подготовки ответа.

9. Методические рекомендации для преподавателей по организации воспитательной работы с обучающимися

Методическое обеспечение процесса воспитания обучающихся выступает одним из определяющих факторов высокого качества образования. Преподаватель вуза, демонстрируя высокий профессионализм, эрудицию, четкую гражданскую позицию, самодисциплину, творческий подход в решении профессиональных задач, в ходе образовательного процесса способствует формированию гармоничной личности.

При реализации дисциплины преподаватель может использовать следующие методы воспитательной работы:

- методы формирования сознания личности (беседа, диспут, внушение, инструктаж, контроль, объяснение, пример, самоконтроль, рассказ, совет, убеждение и др.);

- методы организации деятельности и формирования опыта поведения (задание, общественное мнение, педагогическое требование, поручение, приучение, создание воспитывающих ситуаций, тренинг, упражнение, и др.);

- методы мотивации деятельности и поведения (одобрение, поощрение социальной активности, порицание, создание ситуаций успеха, создание ситуаций для эмоционально-нравственных переживаний, соревнование и др.)

При реализации дисциплины преподаватель должен учитывать следующие направления воспитательной деятельности:

Гражданское и патриотическое воспитание:

- формирование у обучающихся целостного мировоззрения, российской идентичности, уважения к своей семье, обществу, государству, принятым в семье и обществе духовно-нравственным и социокультурным ценностям, к национальному, культурному и историческому наследию, формирование стремления к его сохранению и развитию;

- формирование у обучающихся активной гражданской позиции, основанной на традиционных культурных, духовных и нравственных ценностях российского общества, для повышения способности ответственно реализовывать свои конституционные права и обязанности;

- развитие правовой и политической культуры обучающихся, расширение конструктивного участия в принятии решений, затрагивающих их права и интересы, в том числе в различных формах самоорганизации, самоуправления, общественно-значимой деятельности;

- формирование мотивов, нравственных и смысловых установок личности, позволяющих противостоять экстремизму, ксенофобии, дискриминации по социальным, религиозным, расовым, национальным признакам, межэтнической и межконфессиональной нетерпимости, другим негативным социаль-

ным явлениям.

Духовно-нравственное воспитание:

- воспитание чувства достоинства, чести и честности, совестливости, уважения к родителям, учителям, людям старшего поколения;

- формирование принципов коллективизма и солидарности, духа милосердия и сострадания, привычки заботиться о людях, находящихся в трудной жизненной ситуации;

- формирование солидарности и чувства социальной ответственности по отношению к людям с ограниченными возможностями здоровья, преодоление психологических барьеров по отношению к людям с ограниченными возможностями;

- формирование эмоционально насыщенного и духовно возвышенного отношения к миру, способности и умения передавать другим свой эстетический опыт.

Культурно-просветительское воспитание:

- формирование уважения к культурным ценностям родного города, края, страны;

- формирование эстетической картины мира;

- повышение познавательной активности обучающихся.

Научно-образовательное воспитание:

- формирование у обучающихся научного мировоззрения;

- формирование умения получать знания;

- формирование навыков анализа и синтеза информации, в том числе в профессиональной области.

Физическое воспитание:

- формирование ответственного отношения к своему здоровью, потребности в здоровом образе жизни;

- формирование культуры безопасности жизнедеятельности;

- формирование системы мотивации к активному и здоровому образу жизни, занятиям спортом, культуры здорового питания и трезвости.

Профессионально-трудовое воспитание:

- формирование добросовестного, ответственного и творческого отношения к разным видам трудовой деятельности;

- формирование навыков высокой работоспособности и самоорганизации, умение действовать самостоятельно, мобилизовать необходимые ресурсы, правильно оценивая смысл и последствия своих действий;

Экологическое воспитание:

- формирование экологической культуры, бережного отношения к родной земле, экологической картины мира, развитие стремления беречь и охранять природу.

Объем программы для заочной формы обучения

| Вид учебной работы | Всего часов | Курс |
|--|-------------|------|
| | | 1 |
| ОБЩАЯ ТРУДОЕМКОСТЬ ДИСЦИПЛИНЫ | 180 | 180 |
| КОНТАКТНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ С ПРЕПОДАВАТЕЛЕМ, в том числе: | 37 | 37 |
| Лекционные занятия (Лек) | 4 | 4 |
| Лабораторные занятия (Лаб) | 8 | 8 |
| САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ | 159 | 159 |
| Подготовка к промежуточной аттестации в форме: (экзамен) | 9 | 9 |

Лист внесения изменений

Дополнения и изменения в рабочей программе дисциплины на 20__
/20__ учебный год

В программу вносятся следующие изменения:

1. _____
2. _____
3. _____

*Указываются номера страниц, на которых внесены изменения,
и кратко дается характеристика этих изменений*

Программа одобрена на заседании кафедры-разработчика _____,
протокол № _____

Зав. кафедрой _____

Подпись, дата

Программа одобрена методическим советом института Теплоэнергетики
_____ г., протокол № _____

Председатель методического совета _____
института Теплоэнергетики

Подпись, дата

Согласовано:

Руководитель ОП _____

Подпись, дата



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КГЭУ»)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине

Цифровое проектирование и решение инженерных задач на базе C#

(Наименование дисциплины в соответствии с РУП)

Направление подго-
товки

14.04.01 Ядерная энергетика и теплофизика
(Код и наименование направления подготовки)

Направленность(и) (профиль(и))

Цифровой инжиниринг в атомной
энергетике

(Наименование направленности (профиля) образовательной программы)

Квалификация

Магистр

г. Казань, 2022

Оценочные материалы по дисциплине «Цифровое проектирование и решение инженерных задач на базе С#» – комплект контрольно-измерительных материалов, предназначенных для оценивания результатов обучения на соответствие индикаторам достижения компетенций:

УК-1.2. Вырабатывает стратегию решения поставленной задачи (составляет модель, определяет ограничения, вырабатывает критерии, оценивает необходимость дополнительной информации).

УК-1.3. Формирует возможные варианты решения задач.

ОПК-2.4. Использует современные цифровые технологии для решения научно-технических задач

Оценивание результатов обучения по дисциплине осуществляется в рамках текущего контроля успеваемости, проводимого по балльно-рейтинговой системе (БРС), и промежуточной аттестации.

Текущий контроль успеваемости обеспечивает оценивание процесса обучения по дисциплине. При текущем контроле успеваемости используются следующие оценочные средства: защита лабораторных работ; защиты заданий, выполненных индивидуально или группой обучающихся; контроль выполнения самостоятельной работы обучающихся в устной форме.

Промежуточная аттестация имеет целью определить уровень достижения запланированных результатов обучения по дисциплине за 1 курс 1 семестр. Форма промежуточной аттестации – экзамен.

Оценочные материалы включают задания для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся, разработанные в соответствии с рабочей программой дисциплины.

1. Технологическая карта

Семестр 1

| Наименование контрольного мероприятия | Рейтинговые показатели | | | | Промежуточная аттестация |
|--|------------------------|---------------------|----------------------|-------|--------------------------|
| | I текущий контроль | II текущий контроль | III текущий контроль | Итого | |
| | Итого | | | | |
| Текущий контроль | | | | | |
| Раздел 1. Синтаксис С# | 12 | | | 12 | |
| Тест | 8 | | | 8 | |
| Проектное задание | 4 | | | 4 | |
| Раздел 2. Объектно-ориентированное программирование | | 24 | | 24 | |
| Тест | | 8 | | 8 | |
| Проектное задание | | 12 | | 12 | |
| Кейс-задача | | 4 | | 4 | |
| Раздел 3. Наследование и полиморфизм в ООП | | | 24 | 24 | |
| Тест | | | 8 | 8 | |
| Проектное задание | | | 12 | 12 | |
| Кейс-задача | | | 4 | 4 | |

| | | | | | |
|---------------------------------|--|--|--|-----------|------------|
| Итого за 3 ТК | | | | 60 | |
| Промежуточная аттестация | | | | | |
| Экзамен, в т.ч.: | | | | | 40 |
| – тест | | | | | 20 |
| – проектное задание | | | | | 20 |
| Всего баллов | | | | | 100 |

2. Перечень оценочных средств

Краткая характеристика оценочных средств, используемых при текущем контроле успеваемости и промежуточной аттестации обучающегося по дисциплине:

| Наименование оценочного средства | Краткая характеристика оценочного средства | Оценочные материалы |
|----------------------------------|---|--|
| Кейс-задача (КЗ) | Проблемное задание, в котором обучающемуся предлагается выполнить рефакторинг реально существующего проекта. Задание представляет из себя нестандартную задачу, требующую творческого подхода от исполнителя, всецело измеряющую его трудовые функции. | Комплект кейс-заданий |
| Проектное задание (ПЗ) | Комплект стандартизированных заданий, включающих систему юнит-тестов, для автоматизированного измерения качества проекта обучающегося. Позволяет всесторонне оценить умения слушателя, его способность выполнять свои трудовые функции | Комплект заданий для лабораторных работ |
| Тест (Тест) | Комплект из 120 тестовых вопросов, размещенных на платформе Moodle2. Оценочное средство позволяет оценить глубину теоретических знаний слушателя, частично оценить его умения и проверить трудовые функции. Комплект включает 80% открытых и 20% закрытых заданий | Комплект тестовых вопросов в системе Moodle2 |

3. Оценочные материалы текущего контроля успеваемости обучающихся

Характеристика всех оценочных материалов текущего контроля успеваемости обучающихся в соответствии с технологической картой и перечнем оценочных средств по дисциплине

| | |
|--|---|
| Наименование оценочного средства | Кейс-задача (КЗ) |
| Представление и содержание оценочных материалов | Требуется выполнить рефакторинг кода функционирующих проектов: 1) Incapsulation.Photoshop (раздел «Инкапсуляция») 2) Inheritance.Photoshop (разделы «Наследование», «Полиморфизм») |
| Критерии оценки и шкала оценивания в баллах ² | Проект Incapsulation.Photoshop 1) Выполнен рефакторинг с выделением класса Pixel – 1 балл 2) Выполнена защита целостности полей класса Pixel – 1 балл 3) К классу фото добавлен индекатор, возвращающий / устанавливающий пиксель по переданным координатам – 1 балл 4) Выполнена перегрузка оператора умножения – 1 балл |

² В соответствии с БРС, поддерживаемой преподавателем в ЭИОС

| | |
|---|---|
| | <p>Количество баллов: максимум – 4</p> <p>Проект Inheritance.Photoshop</p> <p>1) Создан класс GrayscaleFilter, реализующий интерфейс IFilter, и переводящий изображение в черно-белую гамму – 2 балла</p> <p>2) Выделен метод ProcessPixel, обрабатывающий один пиксель – 1 балл</p> <p>3) Выделен абстрактный базовый класс PixelFilter. ProcessPixel получился абстрактным методом – 1 балл</p> <p>Количество баллов: максимум – 4</p> |
| Наименование оценочного средства | Проектное задание (ПЗ) |
| Представление и содержание оценочных материалов | <p>Автоматизированные программные тесты для лабораторных работ:</p> <p>1 раздел Beginer.IRR (раздел «Знакомство с синтаксисом C#»)</p> <p>2 раздел Incapsulation.Failures (раздел «Инкапсуляция») Incapsulation.RationalNumbers (раздел «Инкапсуляция») Incapsulation.Weights (раздел «Инкапсуляция»)</p> <p>3 раздел Inheritance.DataStucture (раздел «Наследование») Inheritance.Geometry (раздел «Наследование») Inheritance.MapObjects (раздел «Полиморфизм»)</p> |
| Критерии оценки и шкала оценивания в баллах | <p>– Пройдены все тесты при оптимальной архитектуре проекта – 4 балла</p> <p>– Пройдены все тесты при дефектной архитектуре проекта (использованы т.н. «костыли» для прохождения тестов) – 3 баллов</p> <p>– Пройден хотя бы один тест, некорректная архитектура – 2 балла</p> <p>– Проект скомпилировался и запустился, но тесты не пройдены – 1 балл</p> <p>– Проект не скомпилировался и не запустился – 0 баллов.</p> <p>Количество баллов за ОДНУ работу: максимум – 4</p> |
| Наименование оценочного средства | Тест (Тест) |
| Представление и содержание оценочных материалов | <p>Тест содержит 8 открытых (80%) и закрытых (20%) вопросов. Тестирование подразумевает использование компьютерной техники и платформы moodle2.</p> <p style="text-align: center;"><i>Пример тестового задания</i></p> <p>1. Имеют ли в C# общего предка (в терминах наследования классов) типы System.String и short?</p> <ul style="list-style-type: none"> • Да • Нет <p>2. Что выведет в консоль этот код:</p> <pre>public static class Program { delegate void Action(); public static void Main() { short foo = 0; Action code = delegate { foo++; }; } }</pre> |

```
};  
  
foo++;  
code();  
System.Console.Write(foo);  
}  
}
```

- 1
- 3
- Не скомпилируется
- 0
- 2
- Будет выброшено исключение
- Поведение кода не определено (undefined behaviour)

3. В каких областях памяти могут размещаться объекты значимых (англ. value) типов?

- Heap
- Stack

4. Что выведет в консоль этот код:

```
public static class Program  
{  
    public static void Main()  
    {  
        int num = 594 / 100;  
        int[] arr = {num, 1207 % 5};  
  
        foreach(int Num in arr)  
            System.Console.Write(Num);  
    }  
}
```

- 55
- Не скомпилируется
- 52
- 5
- Будет выброшено исключение

5. Какие из этих утверждений верны для абстрактных базовых классов, но не верны для интерфейсов

- Могут содержать реализацию методов базового класса или интерфейса
- Могут содержать приватные члены
- Могут содержать поля
- Могут содержать методы с реализованной логикой
- Могут содержать свойства

6. Имеется код:

```
interface A { }  
interface B : A { }  
  
class X : A { }  
class Y : X, B { }  
class Z : X { }
```

Какие операторы не выбросят исключение:

| | |
|---|--|
| | <ul style="list-style-type: none"> • B(Z) • A(Y) • A(X) • X as B • Y as B <p>7. Для каких типов и в каком случае возможно неявное преобразование в C#?</p> <ul style="list-style-type: none"> • Для арифметических типов во всех случаях • При преобразовании ссылки на объект одного класса на объект другого класса • При преобразовании ссылки на объект базового класса на объект производного класса • При преобразовании ссылки на объект производного класса на объект базового класса • C# язык строгой типизации, поэтому неявное преобразование типов невозможно • Для арифметических типов, в случае, когда оно не ведет к потере точности данных <p>8. Имеется код:</p> <pre> class steam { private double pressure; private double temp; private bool overheated; public steam(double pressure, double temp, bool overheated) { this.pressure = pressure; this.temp = temp; this.overheated = overheated; } public steam(double pressure, double temp) { this.pressure = pressure; this.temp = temp; } } </pre> <p>Что означает дублирование метода <code>public steam</code>?</p> <ul style="list-style-type: none"> • Перегрузка метода внутри класса для разного количества аргументов • Перегрузка конструктора класса под разное количество полей • Перегрузка свойства полей класса • Ошибку в программе |
| Критерии оценки и шкала оценивания в баллах | <p>Каждый верный ответ на <u>один</u> тестовый вопрос даёт обучающемуся 1 балл</p> <p>Максимальное количество баллов за тест – 8</p> |

4. Оценочные материалы промежуточной аттестации

Дается характеристика всех оценочных материалов промежуточной аттестации обучающихся в соответствии с технологической картой дисциплины

| | |
|----------------------------------|---------|
| Наименование оценочного средства | Экзамен |
|----------------------------------|---------|

Оценочные материалы, вынесенные на экзамен, состоят из теста на проверку теоретических знаний, и задачи для проверки практических умений.

Тест содержит 20 открытых (80%) и закрытых (20%) вопросов. Тестирование подразумевает использование компьютерной техники и платформы moodle2.

Пример тестового задания:

1. Что выведет в консоль метод Console.WriteLine?

```
class Company
{
    private int persons;

    public int Persons
    {
        set { persons = value; }
    }
}

internal class Program
{
    static void Main()
    {
        Company company = new Company() { Persons = 10 };
        Console.WriteLine(company.Persons);
    }
}
```

2. Изучите следующий код:

```
class ClassA
{
    public string FieldA;
}

class ClassB : ClassA
{
    public string FieldB;
}

class ClassC : ClassA
{
    public string FieldC;
}

class ClassD : ClassC
{
    public string FieldC;
}
```

Какие поля содержатся в классе **ClassC**? Перенесите маркер(ы) в область правильного ответа

| | | | |
|---------|---------|---------|---------|
| Field A | Field B | Field C | Field D |
|---------|---------|---------|---------|

3. Заполните пропуск в определении:

{Деструктор/конструктор/поле/свойство} – специализированный метод класса, предназначенный для инициализации объекта при его создании

Представление и содержание оценочных материалов

4. Какие методы должны быть описаны в классе C?

```
interface A
{
    void method1();
    void method2();
}

interface B : A
{
    void method3();
    void method4();
}

interface C : B
{
}
```

Выберите один ответ:

- a. Только методы интерфейса B
- b. Методы интерфейсов A и B
- c. Только методы интерфейса A
- d. Ни методы A, ни методы B

5. Изучите следующий код:

```
class ClassA
{
    public string FieldA;
}

class ClassB : ClassA
{
    public string FieldB;
}

class ClassC : ClassA
{
    public string FieldC;
}

class ClassD : ClassC
{
    public string FieldC;
}
```

Какие поля содержатся в классе **ClassA**? Перенесите маркер(ы) в область правильного ответа

| | | | |
|---------|---------|---------|---------|
| Field A | Field B | Field C | Field D |
|---------|---------|---------|---------|

6. Что будет выведено в консоль при вызове метода H()?

```
class Sample2
{
    static string who = "class";

    static void F() { string who = "F"; }

    static void G()
    {
        F();
        Console.WriteLine(who);
    }
}
```

```

static void H()
{
    string who = "H";
    F();
    Console.WriteLine(who);
}
}

```

В случае ошибки, укажите в ответе **ОШИБКА**

7. Какая из карт памяти соответствует коду к моменту выхода из метода **Update**?

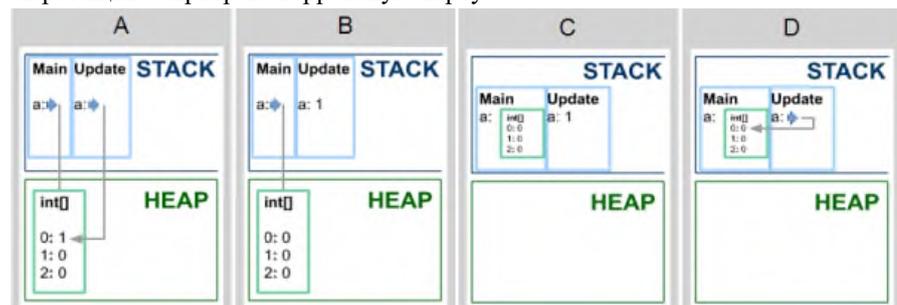
```

class Program
{
    static void Update(int a)
    {
        a++;
        // ?
    }

    static void Main()
    {
        int[] a = new int[3];
        Update(a[0]);
    }
}

```

Перетащите маркер на корректную карту



8. Имеется код:

```

class steam
{
    private double pressure;
    private double temp;
    private bool overheated;

    public steam(double pressure, double temp, bool overheated)
    {
        this.pressure = pressure;
        this.temp = temp;
        this.overheated = overheated;
    }

    public steam(double pressure, double temp)
    {
        this.pressure = pressure;
        this.temp = temp;
    }
}

```

Что означает дублирование метода **public steam**?

Выберите один ответ:

- a. Перегрузка конструктора класса под разное количество полей
- b. Перегрузка метода внутри класса для разного количества аргументов
- c. Ошибку в программе
- d. Перегрузка свойства полей класса

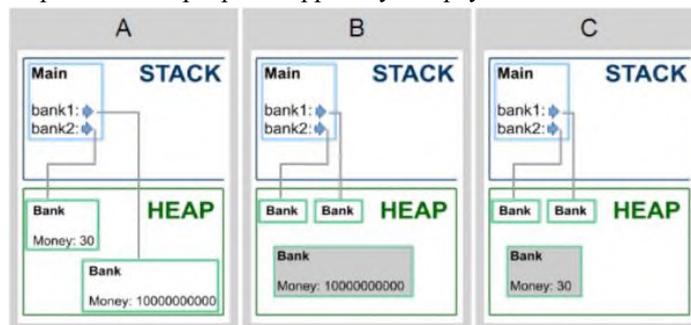
9. Какая из карт памяти соответствует коду к моменту выхода из метода **Main**?

```
class Bank
{
    public static long Money;

    public void SetMoney(long x) { Money = x; }
}

class Program
{
    static void Main()
    {
        var bank1 = new Bank();
        var bank2 = new Bank();
        bank1.SetMoney(10000000000);
        bank2.SetMoney(30);
        //?
    }
}
```

Перетащите маркер на корректную карту

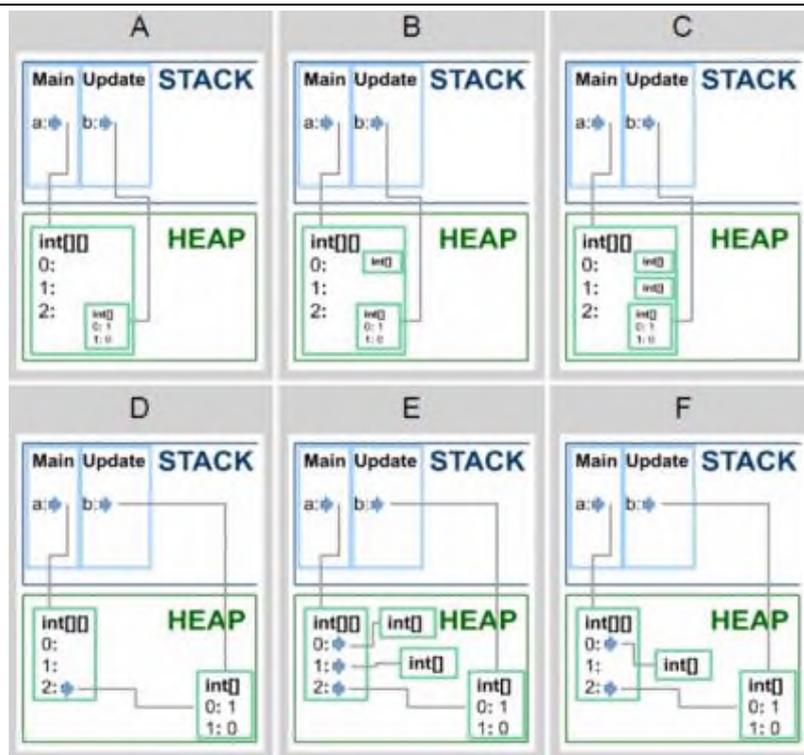


10. Какая из карт памяти соответствует коду к моменту выхода из метода Update?

```
class Program
{
    static void Update(int[] b)
    {
        b[0]++;
        //?
    }

    static void Main()
    {
        var a = new int[3][];
        a[0] = new int[0];
        a[2] = new int[2];
        Update(a[2]);
    }
}
```

Перетащите маркер на корректную карту



11. Чему равна переменная **summa** во время выхода из цикла **for**?

```
class Program
{
    static void Main()
    {
        int num = 9, summa = 3;

        for (int i = 0; i < num; ++i)
            summa += i++;
    }
}
```

12. Какой результат выдаст программа при запуске?

```
public class Substance
{
    public double T = 300;
    public double G;
}

public class Program
{
    public static void Main()
    {
        Console.WriteLine(new Substance() { G = 10 }.T + 15);
    }
}
```

В случае ошибки, укажите **ОШИБКА**

13. Какая из карт памяти соответствует коду к моменту выхода из метода **Main**?

```
class Box { public int Content; }

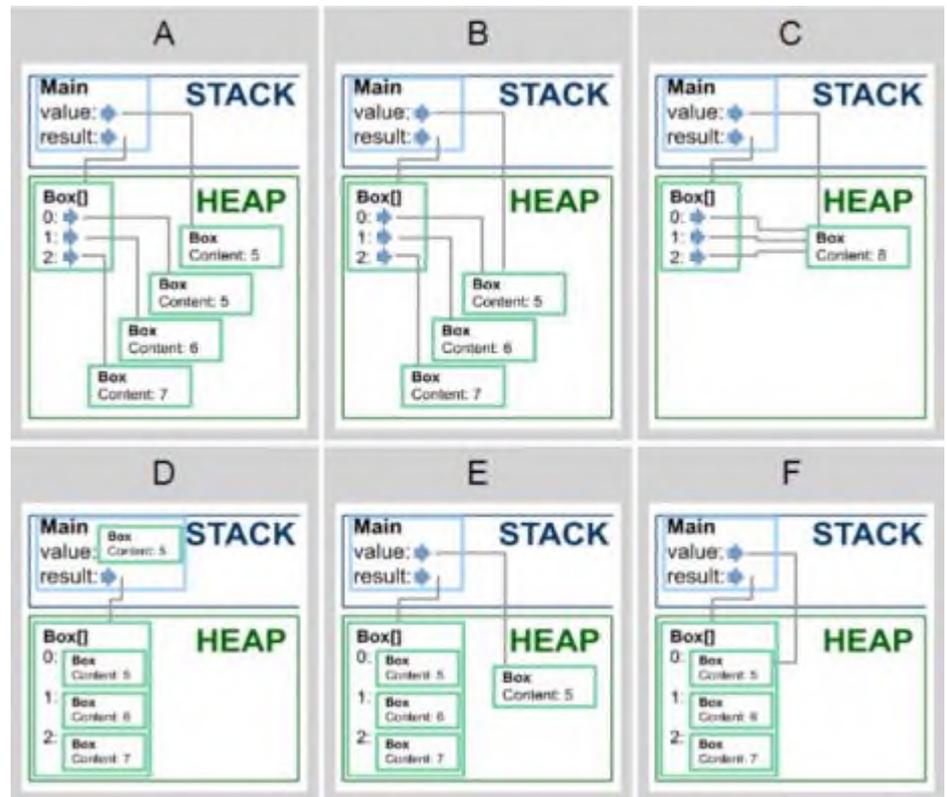
public class Program
{
    static void Main()
    {
        var value = new Box { Content = 5 };
    }
}
```

```

var result = new Box[3];
for (int i = 0; i < result.Length; i++)
{
    result[i] = value;
    value.Content++;
}
//?
}

```

Перетащите маркер на корректную карту



14. Что будет выведено в консоль в результате выполнения программы?

```

public class Program
{
    static void Main()
    {
        int[][] array =
        {
            new int[] { 4, 2, 11, 5},
            new int[] { 5, 3, 1, 0}
        };
        Console.WriteLine(diffArray(array, 2));
    }

    public static string diffArray(int[][] arrays, int index)
    {
        if (arrays[0].Length != arrays[1].Length)
            return "array sizes do not match";

        var result = new int[arrays[0].Length];

        for (int i = 0; i < arrays[0].Length; i++)
            result[i] = arrays[0][i] - arrays[1][i];

        return result[index].ToString();
    }
}

```

```
}  
}
```

В случае ошибки, укажите **ОШИБКА**

15. Что означает оператор == в языке C#?

Выберите один ответ:

- a. оператор == проверяет, представляют ли два разных объекта одно значение
- b. оператор == присваивает значение одного операнда другому операнду
- c. оператор == проверяет, являются ли операнды одним объектом

16. К какому классу относятся методы WriteLine() и Read()?

17. Что будет выведено в консоль?

```
class Program  
{  
    public static void Main()  
    {  
        float real = 0.0f / 0.0f;  
        if (real == real) Console.WriteLine("Equal");  
        else Console.WriteLine("Not equal");  
    }  
}
```

В случае ошибки, напишите в блоке ответа **ОШИБКА**

18. Какой результат будет выведен в консоль в результате выполнения программы?

```
public static class Program  
{  
    public static void Main()  
    {  
        int x = 1, value = 500;  
        double a = 3;  
        while (a < value)  
        {  
            a += Math.Pow(a, x++);  
        }  
        Console.WriteLine(x);  
    }  
}
```

19. Заполните пропуск в определении:

{...} – специализированный метод класса, предназначенный для инициализации объекта при его создании

20. Что будет выведено в консоль при вызове метода G()?

```
class Sample2  
{  
    static string who = "class";  
  
    static void F() { string who = "F"; }  
  
    static void G()  
    {  
        F();  
        Console.WriteLine(who);  
    }  
  
    static void H()  
    {  
        string who = "H";  
    }  
}
```

| | |
|--|--|
| | <pre> F(); Console.WriteLine(who); } } </pre> <p>В случае ошибки, укажите в ответе ОШИБКА</p> <p><i>Пример практического задания:</i></p> <p>В результате эксперимента получены наборы значений аргумента X и соответствующих значений функции Y. Необходимо сформировать таблицу значений функции, упорядочив их по возрастанию X. Если одному значению X соответствует несколько значений Y, взять их среднее значение.</p> <p>В файле table.cs проекта Beginer.Table напишите функцию:</p> <pre> public static double[,] SortTable(List<Table> stockTable) </pre> <p>принимающую на вход наборы значений аргумента X и соответствующих значений функции Y, а возвращающая упорядоченный двумерный массив типа double: Y(X).</p> |
| <p>Критерии оценки и шкала оценивания в баллах</p> | <p>Каждый верный ответ на <u>один</u> тестовый вопрос даёт обучающемуся 1 балл</p> <p>Максимальное количество баллов за тест – 20</p> <p>Критерии оценки практического задания:</p> <ul style="list-style-type: none"> – Пройдены все тесты при оптимальной архитектуре проекта – 20 баллов – Пройдены все тесты при дефектной архитектуре проекта (использованы т.н. «костыли» для прохождения тестов) – 15 баллов – Пройден хотя бы один тест, некорректная архитектура – 10 баллов – Проект скомпилировался и запустился, но тесты не пройдены – 5 балла – Проект не скомпилировался и не запустился – 0 баллов. <p>Максимальное количество баллов за выполнение практического задания – 20</p> <p>Максимальное количество баллов за экзамен - 40</p> |