

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шебзухова Татьяна Александровна
Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета
Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования

Дата подписания: 27.05.2025 16:28:23

«Северо-Кавказский федеральный университет»

Уникальный программный ключ:

Пятигорский институт (филиал) СКФУ

d74ce93cd40e39275c3ba2f58486412a1c8cf866f

Колледж Пятигорского института (филиал) СКФУ

УТВЕРЖДАЮ

Директор Пятигорского института
(филиал) СКФУ
Т.А. Шебзухова

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

По профессиональному модулю

ПМ.02 «Осуществление интеграции программных
модулей»

Специальность

09.02.07 «Информационные системы и
программирование»

Форма обучения

очная

Пятигорск 2025г.

Фонд оценочных средств по профессиональному модулю ПМ.02 «Осуществление интеграции программных модулей» разработан на основании федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 «Информационные системы и программирование».

Разработчик: преподаватель колледжа Пятигорского института (филиал) СКФУ,
Антоненко Дмитрий Сергеевич

СОГЛАСОВАНО:

Представитель работодателя

Зам. Генерального директора
ООО «Миллениум - плюс»

должность представителя работодателя, наименование
организации и город ее расположения

подпись

Давыдов А.А.

Фамилия, инициалы

М.П.

1. Паспорт фонда оценочных средств

1.1. Область применения

Фонд оценочных средств (далее - ФОС) предназначен для проверки результатов освоения вида деятельности (ВД) ПМ.02 Осуществление интеграции программных модулей и составляющих его профессиональных и общих компетенций, образовательной программы СПО по профессии/специальности 09.02.07 «Информационные системы и программирование».

ФОС разработан на основании ФГОС, образовательной программы СПО и рабочей программы профессионального модуля (далее - ПМ).

1.2. Планируемые результаты освоения профессионального модуля

Результатом освоения ПМ является готовность обучающегося к выполнению вида профессиональной деятельности (в соответствии с рабочей программой ПМ) и сформированность профессиональных и общих компетенций.

Формой аттестации по ПМ является экзамен по модулю.

2. Формы контроля и оценивания элементов профессионального модуля

Таблица 1

Элемент профессионального модуля	Форма контроля и оценивания	
	Промежуточная аттестация	Текущий контроль
МДК.02.01 «Технология разработки программного обеспечения»	Контрольная работа Экзамен	Контрольный срез
МДК.02.02 «Инструментальные средства разработки программного обеспечения»	Контрольная работа Дифференцированный зачет	Контрольный срез
МДК.02.03 «Математическое моделирование»	Дифференцированный зачет	Контрольный срез
УП	Дифференцированный зачет	
ПМ (в целом)	Экзамен по модулю	

3. Результаты освоения профессионального модуля

3.1. Оценка профессиональных и общих компетенций

В результате контроля и оценки по ПМ осуществляется комплексная проверка следующих профессиональных и общих компетенций:

Таблица 2

Профессиональные компетенции	Показатели оценки результата
ПК 2.1.	Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент.
ПК 2.2.	Выполнять интеграцию модулей в программное обеспечение.
ПК 2.3.	Выполнять отладку программного модуля с использованием

	специализированных программных средств.
ПК 2.4.	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования
Общие компетенции	Показатели оценки результата
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности.
ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях.
ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде.
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

3.2. Общие и (или) профессиональные компетенции, проверяемые дополнительно: (не предусмотрено)

3.3. Требования к портфолио: (не предусмотрено)

3.4. Требования к курсовой работе (проекту): (не предусмотрено)

4. Оценка освоения теоретического курса профессионального модуля

4.1. Оценочные средства текущего контроля успеваемости и критерии оценки

МДК.02.01 Технология разработки программного обеспечения

Вопросы к контрольным срезам 3 семестр

Вариант 1

1. Что такое технология разработки ПО?
2. Назовите основные особенности и стадии «Спиральной модели».
3. Преобразование целых чисел

Вариант 2

1. Назовите основные особенности и стадии «Каскадной модели».
2. Чем отличаются программа и программное обеспечение?
3. Преобразование строки в число

Вариант 3

1. Понятие жизненного цикла ПО
2. Методология Scrum. Какие группы ролей определены в данной методологии?

3. Ввод данных в C#. Вывод данных в C#

Вариант 4

1. Что понимается под процессом жизненного цикла?
2. Назовите существующие гибкие методологии разработки ПО.
3. Типы данных C#

Вопросы к контрольным срезам 4 семестр

Вариант 1

1. Основная цель процесса анализа требований к программным средствам
2. Назовите существующие модели ЖЦ ПО.
3. Ключевые слова C#

Вариант 2

1. Чем модель ЖЦ ПО отличается от методологии разработки ПО?
2. Преобразование действительных чисел
3. Преобразование чисел в строку

Вариант 3

1. Что понимается под моделью ЖЦ ПО?
2. Инкремент и декремент
3. Логические операторы

Вопросы к контрольным работам 3 семестр

Вариант 1

1. Понятие технологии разработки ПО?
2. Язык C#, типы данных и синтаксис
3. Напишите программу для расчета квадратного уравнения

Вариант 2

1. Жизненный цикл программного обеспечения
2. Понятие ООП. Основные принципы ООП
3. Напишите программу для расчета площади треугольника

Вариант 3

1. Методологии разработки программного обеспечения
2. Создание классов в C#
3. Напишите программу для расчета площади параллелепипеда.

Критерии оценивания компетенций

1. Оценка «5» (отлично) выставляется в случае полного выполнения контрольной работы, отсутствия ошибок, грамотного текста, точность формулировок и т.д.;
2. Оценка «4» (хорошо) выставляется в случае полного выполнения всего объема контрольной работы при наличии несущественных ошибок, не повлиявших на общий результат работы и т.д.;
3. Оценка «3» (удовлетворительно) выставляется в случае недостаточно полного выполнения всех разделов контрольной работы, при наличии ошибок, которые не оказали существенного влияния на окончательный результат, при очень ограниченном объеме используемых понятий и т.д.;
4. Оценка «2» (неудовлетворительно) выставляется в случае, если допущены принципиальные ошибки, контрольная работа выполнена крайне небрежно и т.д.

Вопросы к экзамену МДК.02.01

1. Понятие программного обеспечения.
2. Понятие технологии разработки программного обеспечения
3. Жизненный цикл программного обеспечения
4. Спиральная модель разработки
5. Каскадная модель разработки.
6. Качество программного обеспечения
7. Методологии разработки программного обеспечения
8. Методология RUP
9. Методология «Экстремальное программирование»
10. Методология Cristal Clear
11. Методология Scrum
12. Понятие ООП
13. Основные принципы ООП
14. Методы выявления требований к по. Уровни требований
15. Анализ требований к по
16. Стили программирования
17. Императивные и декларативные языки
18. Функциональные и нефункциональные требования
19. Методы выявления требований.
20. Аналитик требований. Основные задачи аналитика требований.

Критерии оценивания компетенций

Оценка «отлично» выставляется студенту за глубокое и полное овладение содержанием учебного материала, в котором студент легко ориентируется, владение понятийным аппаратом за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логичное изложение ответа.

Оценка «хорошо» выставляется студенту, если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ, но содержание и форма ответа имеют некоторые неточности.

Оценка «удовлетворительно» выставляется студенту, если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения практических задач, не умеет доказательно обосновать свои суждения.

Оценка «неудовлетворительно» выставляется студенту, если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажает их смысл, беспорядочно и неуверенно излагает материал, не может применять знания для решения практических задач; за полное незнание и непонимание учебного материала или отказ отвечать.

4.2 МДК.02.02 Инструментальные средства разработки программного обеспечения

Комплект заданий для контрольной работы

1. UML – средства описания проекта на логической стадии разработки.
2. Базы данных. Основные понятия.
3. Базы знаний.
4. Виды инструментальных средств.
5. Диаграмма взаимодействия ПО, как способ выражение сценария ПО.
6. Диаграмма классов: структура, состав, связи.
7. Диаграмма компонентов для объектно-ориентированной системы и web-системы.
8. Диаграмма коопераций: определение, идеология, структура, пример.
9. Диаграмма последовательностей: определение, структура, состав, пример.
10. Диаграмма развертывания и архитектура ПО: сходство и отличие.
11. Диаграмма развертывания: назначение, структура, пример.
12. Диаграмма состояний: определение, назначение, структура, пример.
13. Инсталляция и установка программных систем – проблемы, пути решения, инструменты.
14. Информационный поиск. Модели поиска. Стратегии поиска.
15. Классификация направлений программирования и их особенностей.
16. Классификация стандартов программирования.
17. Логическая форма графического описания взаимодействия активных объектов системы.
18. Методы разработки программы.
19. Глобальные и локальные, статические и динамические, внутренние и внешние методы и инструменты реализации.
20. Перспективы развития инструментальных средств.
21. Полнофункциональность и целостность ПО.
22. Понятие концептуальной, логической, физической структуры БД.
23. Понятие модели данных.
24. Последовательность действий при разработке программ.

25. Построение контекстной помощи – средства и методики.
26. Современные языки программирования ПО.

Критерии оценивания компетенций

Оценка «отлично» выставляется студенту за глубокое и полное овладение содержанием учебного материала, в котором студент легко ориентируется, владение понятийным аппаратом, за умение применять теоретические знания при решении практических задач. Отличная отметка предполагает грамотное, логичное изложение ответа (в письменной форме), качественное внешнее оформление.

Оценка «хорошо» выставляется студенту, если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ в письменной форме, но содержание и форма ответа имеют некоторые неточности.

Оценка «удовлетворительно» выставляется студенту, если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности, при выполнении практических заданий.

Оценка «неудовлетворительно» выставляется студенту, если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает грубые ошибки, при выполнении практических заданий, не может применять знания для решения практических заданий; за полное незнание и непонимание учебного материала или отказ от выполнения письменной работы.

Вопросы для контрольных срезов 4,5 семестр

Контрольный срез 1

Вариант 1

1. Какой из операторов используется для создания цикла в Python?

- а) if
- б) for
- в) def
- г) import

2. Что вернёт функция `len([1, 2, 3, 4])`?

- а) 3
- б) 4
- в) 5
- г) Ошибку

3. Какой тип данных является неизменяемым в Python?

- а) list
- б) tuple
- в) dict
- г) set

4. Как в Python обозначается комментарий?

- а) //
- б) #
- в) /* */
- г) @

Вариант 2

1. Как создать функцию в Python?

- а) function имя_функции():
- б) def имя_функции():
- в) func имя_функции():
- г) lambda имя_функции():

2. Что делает выражение return внутри функции?

- а) Завершает выполнение функции и возвращает значение
- б) Ничего не делает
- в) Выполняет цикл
- г) Импортирует модуль

3. Какой модуль необходимо импортировать для работы со случайными числами?

- а) random
- б) math
- в) os
- г) time

4. Как импортировать только одну функцию из модуля? Например, sqrt из math.

- а) import math.sqrt
- б) from math import sqrt
- в) import sqrt from math
- г) from sqrt import math

Контрольный срез 2

Вариант 1

1. Какой оператор используется для создания условной конструкции в Python?

- а) loop
- б) if

- в) switch
- г) case

2. Как записать условие проверки равенства в Python?

- а) =
- б) ==
- в) :=
- г) ===

3. Какой из методов прерывает выполнение цикла?

- а) stop
- б) return
- в) break
- г) exit

4. Что вернёт следующий код?

```
x = 5
if x > 3:
    print("Большой")
else:
    print("Маленький")
```

- а) Большой
- б) Маленький
- в) Ошибка
- г) Ничего

Вариант 2

1. Какой метод используется для преобразования строки к верхнему регистру?

- а) toUpperCase()
- б) upper()
- в) to_upper()
- г) capitalize()

2. Что делает метод .strip() у строки?

- а) Заменяет пробелы на дефисы
- б) Удаляет пробелы в начале и конце строки
- в) Обрезает строку до определённой длины
- г) Переводит строку в нижний регистр

3. Что вернёт выражение 'Python'.lower()?

- а) python
- б) PYTHON
- в) Ошибку
- г) pYTHON

4. Как объединить список строк ['a', 'b', 'c'] в одну строку через запятую?

- а) join(['a', 'b', 'c'])
- б) ','.join(['a', 'b', 'c'])
- в) ''.concatenate(['a', 'b', 'c'])
- г) concat(['a', 'b', 'c'])

Контрольная работа

1. Что вернёт следующий код?

```
a = [1, 2, 3]
a.append(4)
print(a)
```

- а) [1, 2, 3, 4]
- б) [1, 2, 3]
- в) Ошибку
- г) [1, 2, 3, [4]]

2. Какой результат вернёт выражение 5 // 2?

- а) 2.5
- б) 2
- в) 3
- г) Ошибку

3. Какое из перечисленных утверждений о списках верно?

- а) Они неизменяемы
- б) Они поддерживают индексацию
- в) Они не могут содержать строки
- г) Они всегда имеют фиксированный размер

4. Что делает метод .pop() у списка?

- а) Удаляет и возвращает последний элемент списка
- б) Удаляет элемент по индексу, переданному в аргументе
- в) Добавляет элемент в конец списка
- г) Удаляет все элементы списка

Критерии оценивания компетенций

Оценка «отлично» выставляется студенту за глубокое и полное овладение содержанием учебного материала, в котором студент легко

ориентируется, владение понятийным аппаратом за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логичное изложение ответа.

Оценка «хорошо» выставляется студенту, если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ, но содержание и форма ответа имеют некоторые неточности.

Оценка «удовлетворительно» выставляется студенту, если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения практических задач, не умеет доказательно обосновать свои суждения.

Оценка «неудовлетворительно» выставляется студенту, если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажает их смысл, беспорядочно и неуверенно излагает материал, не может применять знания.

3. Оценочные средства для промежуточной аттестации и критерии оценки:

Процедура дифференцированного зачета как отдельное контрольное мероприятие не проводится, оценивание знаний обучающегося происходит по результатам текущего контроля

4.3 МДК 02.03 Математическое моделирование Темы рефератов

Тема 2. Математические модели, принципы их построения, виды моделей. Инструментарий программирования.

1. Математические модели, принципы их построения.
2. Математические модели, виды моделей.
3. Инструментарий программирования.

Тема 7. Основные понятия динамического программирования: шаговое управление, управление операцией в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий.

1. Основные понятия динамического программирования: шаговое управление, управление операцией в целом.
2. Основные понятия динамического программирования: оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию.
3. Основные понятия динамического программирования: аддитивный критерий, мультипликативный критерий.

Критерии оценивания компетенций

Оценка «отлично» ставится, если выполнены все требования к написанию и защите реферата: обозначена проблема и обоснована её актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объём,

соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы.

Оценка «хорошо» – основные требования к реферату и его защите выполнены, но при этом допущены недочёты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объём реферата; имеются упущения в оформлении; на дополнительные вопросы при защите даны неполные ответы.

Оценка «удовлетворительно» – имеются существенные отступления от требований к реферированию. В частности, тема освещена лишь частично; допущены фактические ошибки в содержании реферата или при ответе на дополнительные вопросы; во время защиты отсутствует вывод.

Оценка «неудовлетворительно» – тема реферата не раскрыта, обнаруживается существенное непонимание проблемы.

Вопросы для контрольных срезов

3 семестр

№ 1

1. Понятие решения, примеры?
2. Множество решений, оптимальное решение примеры?
3. Показатель эффективности решения, примеры?
4. Математические модели, примеры?
5. Принципы построения математических моделей?

№2

1. Общий вид и основная задача линейного программирования.
2. Метод потенциалов
3. Простейшие задачи, решаемые методом динамического программирования.
4. Симплекс-метод решения задач линейного программирования.
5. Интерполяция и предсказание

Оценка «отлично» выставляется студенту за глубокое и полное овладение содержанием учебного материала, в котором студент легко ориентируется, владение понятийным аппаратом, за умение применять теоретические знания при решении практических задач. Отличная отметка предполагает грамотное, логичное изложение ответа (в письменной форме), качественное внешнее оформление.

Оценка «хорошо» выставляется студенту, если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ в письменной форме, но содержание и форма ответа имеют некоторые неточности.

Оценка «удовлетворительно» выставляется студенту, если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности, при выполнении практических заданий.

Оценка «неудовлетворительно» выставляется студенту, если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает грубые ошибки, при выполнении практических заданий, не может применять знания для решения практических заданий; за полное незнание и непонимание учебного материала или отказ от выполнения письменной работы.

3. Оценочные средства для промежуточной аттестации и критерии оценки:
Процедура дифференцированного зачета как отдельное контрольное мероприятие не проводится, оценивание знаний обучающегося происходит по результатам текущего контроля

Комплект вопросов к экзамену по предметному модулю «Осуществление интеграции программных модулей»

1. Что такое технология разработки ПО?
2. Что явилось предпосылкой становления дисциплины «Технология разработки ПО»? Что явилось причиной стремительного развития ПО?
3. Чем отличаются программа и программное обеспечение?
4. Достаточно ли при работе над проектом большой программной системы быть компетентным в области вычислительной техники и программировании. Почему?
5. Может ли большая программная система быть отлажена до конца и почему?
6. Что представляют собой утилитарные программы?
7. При каких условиях созданный программный комплекс может быть назван программным продуктом?
8. Чем отличаются shareware-программы от freeware-программ?
9. Что такое системное программное обеспечение?
10. Что такое инструментарий технологии программирования?
11. Что такое жизненный цикл ПО?
12. Каковы основные этапы разработки ПО (классический жизненный цикл)?
13. Какие мероприятия выполняются на этапе анализа? Что является результатом этого этапа?
14. Какие мероприятия выполняются на этапе проектирования? Что является результатом этого этапа?
15. Какие мероприятия выполняются на этапе внедрения? Что является результатом этого этапа?
16. Какие мероприятия выполняются на этапе сопровождения? Что является результатом этого этапа? Чем отличается классический жизненный цикл от реального жизненного цикла ПО?
17. Каковы достоинства и недостатки классического жизненного цикла?
18. Каковы достоинства и недостатки макетирования?
19. Какие существуют стратегии разработки ПО? Чем они отличаются?
20. Что представляет собой инкрементная модель?
21. Что представляет собой спиральная модель?
22. Каковы достоинства и недостатки спиральной модели?
23. Чем отличаются спиральная и компонентно-ориентированная модели?
24. Каковы достоинства компонентно-ориентированной модели?
25. Что общего и чем отличаются модели: каскадная, спиральная, MSF?
26. Одинакова ли длительность фаз в MSF и может ли деятельность фазы выходить за границы этой фазы? Почему?
27. Какая фаза является нововведением модели MSF по сравнению со спиральной моделью?
28. Математические модели, принципы их построения.

- 29. Математические модели, виды моделей.
- 30. Инструментарий программирования
- 31. Основные понятия динамического программирования: шаговое управление, управление операций в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий.
- 32. Основные понятия динамического программирования: шаговое управление, управление операций в целом.
- 33. Основные понятия динамического программирования: оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию.
- 34. Основные понятия динамического программирования: ад
- 35. Математические операции в Python
- 36. Структура ветвление в Python
- 37. Работа с циклами в Python
- 38. Работа со строками в Python
- 39. Работа со списками. Операции над списками в Python
- 40. Функции и процедуры в Python

Таблица 2– Ключи к фонду оценочных средств

№	Компетенция	Содержание вопроса	Правильный ответ
	ПК 2.1	Что такое технология разработки ПО?	Технология разработки ПО - это система методов, инструментов и подходов к созданию программного обеспечения.
	ПК 2.1	Назовите основные особенности и стадии «Спиральной модели»	Основные особенности и стадии «Спиральной модели»: итеративный подход, акцент на рисках, стадии: планирование, разработка, оценка рисков, планирование следующей итерации.
	ПК 2.1	Преобразование целых чисел	Преобразование целых чисел: в C# используются операции приведения типов, например, (int) или Convert.ToInt32().
	ПК 2.1	Назовите основные особенности и стадии «Каскадной модели».	Основные особенности и стадии «Каскадной модели»: последовательный подход, акцент на документации, стадии: требования, проектирование, реализация, тестирование, развертывание, обслуживание.
	ПК 2.1	Чем отличаются программа и программное обеспечение?	Программа - это набор инструкций для компьютера, программное обеспечение - это совокупность программ, данных и документации.
	ПК 2.1	Преобразование строки в число	Преобразование строки в число: в C# используются методы Convert.ToInt32(), Convert.ToDouble() и т.д.
	ПК 2.1	Понятие жизненного цикла ПО	Жизненный цикл ПО - это совокупность этапов, через которые проходит программное обеспечение от начала разработки до вывода из эксплуатации
	ПК 2.1	Методология Scrum. Какие группы ролей определены в данной методологии?	Методология Scrum: роли - Product Owner, Development Team, Scrum Master; стадии - спринты, обзоры, ретроспективы.
	ПК 2.1	Ввод данных в C#. Вывод данных в C#	Ввод данных в C#: Console.ReadLine(); вывод данных - Console.WriteLine().
	ПК 2.1	Что понимается под процессом жизненного цикла?	Процесс жизненного цикла ПО - это последовательность этапов, на которых происходит разработка, тестирование, развертывание, обслуживание и удаление программного обеспечения.
	ПК 2.1	Назовите существующие гибкие методологии разработки ПО	Существующие гибкие методологии разработки ПО: Agile, Scrum, Kanban, XP
	ПК 2.1	Типы данных C#	Типы данных C#: Value Types (struct, enum, decimal, bool, byte,

			short, int, long, float, double, char, string) и Reference Types (class, interface, delegate, array).
	ПК 2.1	Основная цель процесса анализа требований к программным средствам	Основная цель процесса анализа требований к программным средствам - понимание и документирование потребностей пользователей.
	ПК 2.1	Назовите существующие модели ЖЦ ПО	Существующие модели ЖЦ ПО: Водопадная модель, Итеративная модель, Spiral Model, V-Model, Agile Model
	ПК 2.1	Ключевые слова C#	Ключевые слова C#: class, interface, struct, enum, new, this, base, virtual, override, abstract, sealed, continue, break, return, goto, yield, checked, unchecked, fixed, ref, out, in, async, await, foreach, in, using, lock, try, catch, finally, throw, if, else, switch, case, default, while, do, for, goto, break, continue, return.
	ПК 2.1	Чем модель ЖЦ ПО отличается от методологии разработки ПО	Модель ЖЦ ПО отличается от методологии разработки ПО тем, что модель описывает этапы жизни ПО, а методология - подходы к его разработке.
	ПК 2.2	Преобразование действительных чисел	Преобразование действительных чисел: в C# используются операции приведения типов, например, (double) или Convert.ToDouble().
	ПК 2.2	Преобразование чисел в строку	Преобразование чисел в строку: в C# используются методы ToString() или Convert.ToString().
	ПК 2.2	Что понимается под моделью ЖЦ ПО	Модель ЖЦ ПО - это представление этапов жизни программного обеспечения, таких как планирование, разработка, тестирование, развертывание и обслуживание.
	ПК 2.2	Инкремент и декремент	Инкремент и декремент - это операции увеличения или уменьшения значения переменной на единицу.
	ПК 2.2	Логические операторы	Логические операторы в C#: && (логическое И), (логическое ИЛИ), ! (логическое НЕ), ^ (логическое ИСКЛЮЧАЮЩЕЕ ИЛИ).
	ПК 2.2	Понятие ООП. Основные принципы ООП	Понятие ООП: объектно-ориентированное программирование - подход к разработке программного обеспечения, основанный на Konzeptes der Klassen und Objekte. Основные принципы ООП:

			наследование, полиморфизм, инкапсуляция, абстракция.
	ПК 2.2	Методологии разработки программного обеспечения	Методологии разработки программного обеспечения - это подходы к созданию программного обеспечения, основанные на определенных принципах и практиках
	ПК 2.2	Создание классов в C#	Создание классов в C#: класс - это шаблон для создания объектов, определяется с помощью ключевого слова class, содержит поля, свойства, методы и события.
Инструментальные средства разработки программного обеспечения			
Контрольная работа			
	ПК 2.1	Что вернёт следующий код? <pre>a = [1, 2, 3] a.append(4) print(a)</pre>	а
	ПК 2.1	Какой результат вернёт выражение 5//2?	б
	ПК 2.1	Какое из перечисленных утверждений о списках верно?	б
	ПК 2.1	Что делает метод .pop() у списка?	а
Математическое моделирование			
	ПК 2.3	Понятие решения, примеры?	Решение - результат, завершающий процесс поиска выхода из ситуации. Примеры: решение задачи, решение головоломки, решение проблемы.
	ПК 2.3	Множество решений, оптимальное решение примеры?	Множество решений - набор всех возможных решений задачи. Оптимальное решение - лучшее из доступных решений по определенному критерию. Примеры: в задаче найти наименьшее число, множество решений - все натуральные числа, оптимальное решение - 1.
	ПК 2.3	Показатель эффективности решения, примеры?	Показатель эффективности решения - критерий, используемый для оценки качества решения. Примеры: время выполнения, затраты, прибыль.
	ПК 2.3	Математические модели, примеры?	Математические модели - представление реальных процессов или ситуаций в виде математических структур. Примеры: модель линейного программирования, модель динамического программирования.
	ПК 2.3	Принципы построения математических моделей?	Принципы построения математических моделей: точность, простота, обоснованность, применимость.
	ПК 2.3	Общий вид и основная задача линейного программирования.	Общий вид линейного программирования: минимизировать линейную функцию, Subject to линейные ограничения. Основная задача - нахождение оптимального решения.

	ПК 2.3	Метод потенциалов	Метод потенциалов - метод решения задач коммивояжера и других задач перемещения.
	ПК 2.3	Простейшие задачи, решаемые методом динамического программирования	Простейшие задачи динамического программирования: задачи с оптимальным подбором, задачи с оптимальным планированием.
	ПК 2.3	Симплекс-метод решения задач линейного программирования.	Симплекс-метод - метод решения задач линейного программирования, основанный на перемещении по симплекс-таблице.
	ПК 2.3	Интерполяция и предсказание	Интерполяция и предсказание - методы построения математических моделей для прогнозирования будущих значений на основе имеющихся данных.