

**Перечень теоретических и практических заданий к
дифференцированному зачету
по УП.1 Учебной практики
(3 курс, 5 семестр 2022-2023 уч. г.)**

Форма контроля: Практическая работа (Сравнение с аналогом)

Описательная часть: выполнить два практических задания

Перечень практических заданий:

Задание №1 Создать модуль, содержащий процедуры и/или функции:

длина окружности;

площадь круга; длина дуги;

площадь сектора.

Написать программу, демонстрирующую работу созданного модуля.

Оценка	Показатели оценки
5	Создан модуль, содержащий процедуры и/или функции: длина окружности; площадь круга; длина дуги; площадь сектора. Написана программа, демонстрирующая работу созданного модуля.
4	Создать модуль, содержащий процедуры и/или функции: длина окружности; площадь круга; длина дуги; Написана программа, демонстрирующая работу созданного модуля.
3	Создать модуль, содержащий процедуры и/или функции: длина окружности; Написана программа, демонстрирующая работу созданного модуля.

Задача №2 Создать модуль, содержащий процедуры и/или функции:

длина окружности;

площадь круга; длина дуги;

площадь сектора.

Написать программу, демонстрирующую работу созданного модуля.

Оценка	Показатели оценки
--------	-------------------

Задача №3 Создать модуль, содержащий процедуры и/или функции:

длина окружности;

площадь круга; длина дуги;

площадь сектора.

Написать программу, демонстрирующую работу созданного модуля.

Оценка	Показатели оценки
--------	-------------------

Задача №4 Проведение тестирования типа «белый ящик» над внутренними структурами и модулями конкретной программы.

Перечислить возможные уровни тестирования. Дать ответ на вопрос: На что направлено модульное тестирование?

Оценка	Показатели оценки
5	Проведено тестирование типа «белый ящик» над внутренними структурами и модулями конкретной программы. Перечислены возможные уровни тестирования. Дан ответ на вопрос: На что направлено модульное тестирование?
4	Проведено тестирование типа «белый ящик» над внутренними структурами и модулями конкретной программы. Перечислены возможные уровни тестирования.
3	Проведено тестирование типа «белый ящик» над внутренними структурами и модулями конкретной программы.

Задача №5 Разработать программу, в которой есть элемент "меню", аккуратно и красиво оформленное, благодаря которому можно попадать в отдельную категорию.

Оценка	Показатели оценки
5	Разработана программа, в которой есть элемент "меню", имеется оформление элемента меню имеется переход в отдельную категорию. Код комментирован.

4	Разработана программа, в которой есть элемент "меню", имеется оформление элемента меню и переход в отдельную категорию.
3	Разработана программа, в которой есть элемент "меню", имеется переход в отдельную категорию.

Задание №6 Разработать программу, в которой есть элемент "меню", аккуратно и красиво оформленное, благодаря которому можно попадать в отдельную категорию.

Оценка	Показатели оценки
--------	-------------------

Задание №7 Провести рефакторинг любой созданной ранее программы. Указать причины для рефакторинга для конкретного случая (причина – фрагмент кода). Сохранить работоспособную версию кода. Оценить производительность кода.

Оценка	Показатели оценки
5	Проведен рефакторинг любой созданной ранее программы. Указаны причины для рефакторинга для конкретного случая (две причины – два фрагмента кода). Код работоспособный. Представлено описание по оцениванию производительности кода.
4	Проведен рефакторинг любой созданной ранее программы. Указаны причины для рефакторинга для конкретного случая (две причины – два фрагмента кода). Код работоспособный.
3	Проведен рефакторинг любой созданной ранее программы. Указаны причины для рефакторинга для конкретного случая (одна причина – один фрагмент кода). Код работоспособный.

Задание №8 Техническое задание на разработку ПО. Документ должен включать следующие разделы:

- введение;
- основания для разработки;
- назначение разработки;
- требования к программе;
- требования к программной документации;

- технико-экономические показатели;
- стадии и этапы разработки;
- порядок контроля и приемки;
- приложения.

Оценка	Показатели оценки
5	Разработано техническое задание на разработку ПО. Документ отражает следующие разделы: - введение; - основания для разработки; - назначение разработки; - требования к программе; - требования к программной документации; - технико-экономические показатели; - стадии и этапы разработки; - порядок контроля и приемки; - приложения.

4	Разработано техническое задание на разработку ПО. Документ отражает следующие разделы: - введение; - основания для разработки; - назначение разработки; - требования к программе; - требования к программной документации; - технико-экономические показатели; - стадии и этапы разработки; - порядок контроля и приемки.
3	Разработано техническое задание на разработку ПО. Документ отражает следующие разделы: - введение; - основания для разработки; - назначение разработки; - требования к программе; - требования к программной документации; - технико-экономические показатели; - стадии и этапы разработки.

Задание №9 Алгоритм для задачи:

Сделайте класс `Worker`, в котором будут следующие public поля - `name` (имя), `age` (возраст), `salary` (зарплата).

Создайте объект этого класса, затем установите поля в следующие значения (не в `__construct`, а для созданного объекта) - имя 'Иван', возраст 25, зарплата 1000. Создайте второй объект этого класса, установите поля в следующие значения - имя 'Вася', возраст 26, зарплата 2000.

Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Сделайте класс Worker, в котором будут следующие private поля - name (имя), age (возраст), salary (зарплата) и следующие public методы setName, getName, setAge, getAge, setSalary, getSalary.

Создайте 2 объекта этого класса: 'Иван', возраст 25, зарплата 1000 и 'Вася', возраст 26, зарплата 2000.

Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Дополните класс Worker из предыдущей задачи private методом checkAge, который будет проверять возраст на корректность (от 1 до 100 лет). Этот метод должен использовать метод setAge перед установкой нового возраста (если возраст не корректный - он не должен меняться).

Оценка	Показатели оценки
5	Составлен алгоритм демонстрирующий логически правильную и эффективную программу
4	Составлен алгоритм демонстрирующий логически правильную и эффективную программу. Имеются замечания (одна ошибка)
3	Составлен алгоритм демонстрирующий работу программу.

Продолжение №10
Продолжить работу по созданию классов и объектов на примере задачи (написать программу):

Сделайте класс Worker, в котором будут следующие public поля - name (имя), age (возраст), salary (зарплата).

Создайте объект этого класса, затем установите поля в следующие значения (не в __construct, а для созданного объекта) - имя 'Иван', возраст 25, зарплата 1000. Создайте второй объект этого класса, установите поля в следующие значения - имя 'Вася', возраст 26, зарплата 2000.

Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Сделайте класс Worker, в котором будут следующие private поля - name (имя), age (возраст), salary (зарплата) и следующие public методы setName, getName, setAge, getAge, setSalary, getSalary.

Создайте 2 объекта этого класса: 'Иван', возраст 25, зарплата 1000 и 'Вася', возраст 26, зарплата 2000.

Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Дополните класс Worker из предыдущей задачи private методом checkAge, который будет проверять возраст на корректность (от 1 до 100 лет). Этот метод должен использовать метод setAge перед установкой нового возраста (если возраст не корректный - он не должен меняться).

Оценка	Показатели оценки
--------	-------------------

5	Продемонстрирована работа по созданию классов и объектов на примере задачи
4	Продемонстрирована работа по созданию классов и объектов на примере задачи. Имеются недочеты.
3	Продемонстрирована работа по созданию классов.

Продание №11 Провести интеграционное тестирование на двух или более программных модулей.

Оценка	Показатели оценки
5	Проведена интеграция и тестирование кода каждого функционального компонента. Тестовые варианты покрывают все требования проекта уровня функциональных компонентов ПС.
4	Проведена интеграция и тестирование кода каждого функционального компонента. Тестовые варианты покрывают не все (одно не покрывает) требования проекта уровня функциональных компонентов ПС.
3	Проведена интеграция и тестирование кода каждого функционального компонента. Тестовые варианты покрывают не все (два и более не покрывают) требования проекта уровня функциональных компонентов ПС.

Задание №12 Составить сравнительную характеристику для следующих сред разработки мобильных приложений:

- Visual Studio
- Android Studio
- XCode
- Xamarin Studio

Указать языки программирования.

Указать достоинства и недостатки.

Оценка	Показатели оценки
5	Составлена сравнительная характеристика для следующих сред разработки мобильных приложений: • Visual Studio • Android Studio • XCode • Xamarin Studio Указаны языки программирования. Указаны достоинства и недостатки сред разработки мобильных приложений.

4	Составлена сравнительная характеристика для следующих сред разработки мобильных приложений: • Visual Studio • Android Studio • XCode • Xamarin Studio Указаны языки программирования. Указаны недостатки сред разработки мобильных приложений.
3	Составлена сравнительная характеристика для следующих сред разработки мобильных приложений: • Visual Studio • Android Studio • XCode • Xamarin Studio Указаны языки программирования.

Задание №13 вопросы:

1. Что понимается под понятием «Нативные»? Приведите пример.
2. В чем заключается функциональные особенности адаптированного сайта и приложения?
3. Чем приложение отличается от мобильного сайта?
4. Зачем нужны мобильные приложения?

Оценка	Показатели оценки
5	Даны ответы на все вопросы
4	Даны ответы на первые три вопроса.
3	Даны ответы на первые два вопроса.

Задание №14 вопросы:

Составить алгоритм установки среды для разработки мобильных приложений. Указать особенности при установке, которые необходимо учесть. Назовите представителей лидеров мобильного рынка.

Оценка	Показатели оценки
5	Даны ответы на все вопросы
4	Даны ответы на первые два вопроса.
3	Даны ответы на один вопрос.

Задача №15 используя язык C

1. Разработать программу, которая используя подпрограмму, находит наибольший элемент в массиве чисел (двойных слов). Адрес первого элемента массива и количество элементов в нем передается в качестве параметров в подпрограмму через стек.
2. Разработать программу, которая используя подпрограмму, находит количество четных элементов массива чисел (двойных слов). Адрес первого элемента массива и количество элементов в нем передается в качестве параметров в подпрограмму через стек.
3. Разработать программу, которая используя подпрограмму, находит количество слов в строке. Адрес первого символа строки передается в качестве параметров в подпрограмму через стек. Строка заканчивается нулевым байтом.

Оценка	Показатели оценки
5	Разработано 3 задачи
4	Разработано 2 задачи
3	Разработана 1 задача