

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Попов Анатолий Николаевич
Должность: директор
Дата подписания: 31.05.2023 17:30:49
Уникальный программный ключ:
1e0c38dca0aee73cee1e5e09c1d5873fc7497ba8

Приложение 2
к рабочей программе дисциплины

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Основы и методологии программирования *(наименование дисциплины(модуля))*

Направление подготовки / специальность

27.03.05 Инноватика
(код и наименование)

Направленность (профиль)/специализация

Управление инновациями
(наименование)

Содержание

1. Пояснительная записка.
2. Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций.
3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации.

1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ОПК-7.1 Разрабатывает алгоритмы и программы на языке программирования высокого уровня	ОПК-7.1.1 Знать: <ul style="list-style-type: none">– понятие алгоритма и классификацию алгоритмических языков и систем программирования;– язык программирования высокого уровня - Си;
	ОПК-7.1.2 Уметь: <ul style="list-style-type: none">– применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ;– Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;
	ОПК-7.1.3 Владеть: <ul style="list-style-type: none">– навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач;– навыками разработки программ в водопадной модели жизненного цикла

Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК-7.1 Разрабатывает алгоритмы и программы на языке программирования высокого уровня	ОПК-7.1.1 Знать: <ul style="list-style-type: none">– понятие алгоритма и классификацию алгоритмических языков и систем программирования;– язык программирования высокого уровня - Си;	Тест 1-10
	ОПК-7.1.2 Умеет: <ul style="list-style-type: none">– применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ;– Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;	Задание 1-3
	ОПК-7.1.3 Владеет: <ul style="list-style-type: none">– навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач;– навыками разработки программ в водопадной модели жизненного цикла	Задание 4-7

Промежуточная аттестация (зачет) проводится в одной из следующих форм:

- 1) собеседование;
- 2) выполнение заданий в ЭИОС СамГУПС.

2. Типовые¹ контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций

2.1 Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-7.1.1	Знает: <ul style="list-style-type: none">– понятие алгоритма и классификацию алгоритмических языков и систем программирования;– язык программирования высокого уровня - Си;
<p>1. <i>Инструментальные средства быстрого прототипирования ИС - ...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> языки программирования высокого уровня<input type="checkbox"/> приложения в развитых СУБД – класс DEVELOPER<input type="checkbox"/> интегрированные инструменты быстрой разработки<input type="checkbox"/> приложений – класс BUILDER <p>2. <i>Сущность прототипной (RAD) технологии - ...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> создание на ранней стадии реализации действующей интерактивной модели системы, так называемой системы-прототипа, позволяющей наглядно продемонстрировать пользователю будущую систему, уточнить его требования, оперативно модифицировать интерфейсные элементы<input type="checkbox"/> формализация процесса проектирования ИС<input type="checkbox"/> активное вовлечение конечных пользователей в процесс разработки предназначенных для них интерактивных систем <p>3. <i>К технологиям традиционного прототипирования ИС относятся.....</i></p> <ul style="list-style-type: none"><input type="checkbox"/> демонстрация работы прототипа<input type="checkbox"/> разработка системы прототипа<input type="checkbox"/> доработка системы прототипа<input type="checkbox"/> разработка новых спецификаций требований<input type="checkbox"/> разработка постановки задачи<input type="checkbox"/> документирование готового приложения <p>4. <i>Инструментальными средствами быстрого прототипирования ИС являются ...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> приложения в развитых СУБД - класс DEVELOPER<input type="checkbox"/> интегрированные инструменты быстрой разработки приложений - класс BUILDER<input type="checkbox"/> языки программирования высокого уровня <p>5. <i>Технология итерационного прототипирования ИС - ...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> разработка новой постановки задачи<input type="checkbox"/> демонстрация работы прототипа<input type="checkbox"/> разработка приложения<input type="checkbox"/> доработка системы прототипа<input type="checkbox"/> документирование готового приложения<input type="checkbox"/> разработка системы прототипа <p>6. <i>К основным приемам быстрой разработки прототипа ИС относятся ...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> разработка приложения итерациями<input type="checkbox"/> использование большого количества технических средств<input type="checkbox"/> использование автоматических генераторов (мастеров)<input type="checkbox"/> высокая параллельность работ<input type="checkbox"/> повторное использование частей проекта<input type="checkbox"/> низкая производительность проектировщиков <p>7. <i>Диаграммами, выступающими в качестве инструментальных средств функционально-ориентированного анализа и проектирования являются...</i></p> <ul style="list-style-type: none"><input type="checkbox"/> диаграммы деятельностей (Activitydiagram)<input type="checkbox"/> SSD (SystemStructureDiagram) - диаграмма структуры программного приложения	

¹Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

- диаграммы взаимодействия объектов (Interactiondiagram)
 - диаграммы состояний (Statechartdiagram)
 - DFD (DataFlowDiagram) - диаграмма потоков данных
8. Признаки классификации CASE-средств по режиму коллективной разработки проекта - ...
- не поддерживающие коллективную разработку
 - по виду загруженности разработчиков
 - работающие под управлением WINDOWS
 - ориентированные на режим объединения подпроектов
 - ориентированные на режим реального времени разработки проекта
9. Признаки классификации CASE-средств по типу операционной системы (ОС) работающие под управлением - ...
- WINDOWS 3.11 и выше
 - MicrosoftOffice
 - OS/2 и др.
 - UNIX
10. Диаграммы, выступающие в качестве инструментальных средств объектно-ориентированного анализа и проектирования - ...
- диаграммы компонентов (Componentdiagram)
 - диаграммы деятельностей (Activitydiagram)
 - диаграммы пакетов (Packagediagram)
 - диаграммы размещения (Deploymentdiagram)
 - DFD (DataFlowDiagram) - диаграмма потоков данных
 - диаграммы взаимодействия объектов (Interactiondiagram)

2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-7.1.2	<p>Умеет:</p> <ul style="list-style-type: none"> – применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; – Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;

Задание 1

Написать программы для решения предложенных ниже задач. Пользователь вводит допустимое значение x , программа выдает результат.

1. $y = (1 + \sin x)^2$
2. $y = \sqrt{1 + x^2}$
3. $y = |5 - 6x|$
4. $y = \log_2 \frac{x^2 + 1}{3}$
5. $y = \sqrt[4]{1 + x + x^2}$
6. $y = \sin(\cos x)$
7. $y = \operatorname{ctg}(x - 4)$
8. $y = 2^{x+1}$

Задание 2

1. Вычислить длину окружности, площадь круга и объем шара одного и того же радиуса
2. По координатам трех вершин некоторого треугольника найти его периметр и площадь
3. Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов
4. Вычислить площадь полной поверхности и объем прямого кругового цилиндра, заданного высотой и диаметром основания.

5. Вычислить площадь полной поверхности и объем прямого кругового цилиндра, заданного высотой и диаметром основания.
6. Вычислить периметр и площадь правильного 15-угольника, вписанного в окружность заданного радиуса
7. Вычислить площадь полной поверхности и объем правильного тетраэдра по длине его ребра.
8. Вычислить площадь параллелограмма, у которого даны длины сторон и величина угла между ними (в градусах).

Задание 3

1. Найти максимум из трех заданных целых чисел a, b, c
2. Даны произвольные целые числа a, b, c . Определить, можно ли построить треугольник с длинами сторон, равными этим числам.
3. Найти действительные корни квадратного уравнения $ax^2 + bx + c = 0$. Если корней нет, то вывести сообщение об этом.
4. Решить систему уравнений $y = \begin{cases} ax + by = c \\ dx + ey = f \end{cases}$
5. Дана точка $A(x, y)$. Определить, какой из координатных четвертей она принадлежит.
6. Определить, какая из трех точек $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$ ближе к началу координат.
7. Если треугольник со сторонами a, b , является прямоугольным, то напечатать длину медианы, выходящей из прямого угла, в противном случае напечатать его периметр.
8. Если треугольник со сторонами a, b, c является равнобедренным, найти его площадь, в противном случае найти его периметр.

ОПК-7.1.3

Владеет:

- навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач;
- навыками разработки программ в водопадной модели жизненного цикла

Задание 4

Написать программу, которая запрашивает исходные данные и проводит над ними выбранное пользователем действие. Выбор действия осуществляется с помощью меню.

1. Исходные данные: числа a и b .
меню действий
 1. – сложить
 2. – умножить
 3. – вычесть
 4. – разделить
 5. – выход
 Результат выводить после каждого действия
2. Исходные данные: числа a и b – координаты точки A .
меню действий: шагнуть
 1. вверх
 2. вниз
 3. влево
 4. вправо
 5. выход
 длина шага = 1. Результат после каждого шага.
3. Исходные данные: числа a, b, c – длины сторон прямоугольного параллелепипеда.
меню действий: найти
 1. объем
 2. площадь поверхности
 3. длину диагонали
 4. сумму длин ребер
 5. выход
 результат выводить после каждого действия.

4. Исходные данные: два вектора на плоскости $a(x,y)$ и $b(t,z)$.
меню действий
1. найти сумму векторов
 2. найти разность векторов
 3. найти скалярное произведение векторов
 4. найти сумму длин векторов
 5. выход
- результат выводить после каждого действия
5. меню действий
1. утро
 2. день
 3. вечер
 4. ночь
 5. выход
- времена дня и ночи следуют друг за другом в порядке, указанном в меню. Программа должна выводить время дня и ночи, следующее за выбранным. Результат выводить после каждого действия.
6. Исходные данные: четыре числа a, b, c, d – цены карандаша, ручки, ластика и тетрадки.
меню действий: купить
1. карандаш
 2. ручку
 3. ластик
 4. тетрадь
 5. выход
- после каждого действия нужно выводить общую стоимость выбранных товаров.
7. Исходные данные: две точки на плоскости $X(a, b)$ $Y(c, d)$ – концы отрезка.
меню действий:
1. отразить относительно оси OX
 2. отразить относительно оси OY
 3. отразить относительно точки $(0,0)$
 4. найти длины проекций отрезка на оси OX и OY
 5. выход
- результат выводить после каждого действия.
8. Исходные данные: числа a и b .
меню действий:
1. найти наибольшее
 2. найти наименьшее
 3. найти модуль разности
 4. найти сумму квадратов
 5. выход
- результат выводить после каждого действия

Примечание: следует предусмотреть ввод пользователем некорректных данных

Задание 5

1. Ввести с клавиатуры массив размером N на M , компоненты которого являются действительными числами. Найти сумму квадратов компонент массива.
2. Создать массив размером $N \times M$, компоненты которого являются случайными целыми числами от 0 до MAX . Найти наибольший и наименьший компонент массива.
3. Ввести с клавиатуры массив размером N на M , компоненты которого являются действительными числами. Найти сумму модулей компонент массива.
4. Создать массив размером $N \times M$, компоненты которого являются случайными целыми числами от 0 до MAX . Найти разность наибольшего и наименьшего компонент массива.
5. Ввести с клавиатуры массив размером N на M , компоненты которого являются действительными числами. Найти произведение компонент массива.

6. Создать массив размером $N \times M$, компоненты которого являются случайными целыми числами от 0 до MAX. Найти наибольший компонент массива среди тех, сумма_индексов $I+J$ которых, есть четное число ($A[1,1], A[1,3], \dots, A[2,2], \dots$)
7. Ввести с клавиатуры массив размером N на M , компоненты которого являются действительными числами. Найти сумму синусов от компонент массива.
8. Создать массив размером $N \times M$, компоненты которого являются случайными целыми числами от 0 до MAX. Найти наибольший компонент массива среди элементов, расположенных над главной диагональю (включительно).

Задание 6

1. Напишите программу, подсчитывающую число заданных букв (например Л) в произвольном слове.
2. Напишите программу, удваивающую буквы в произвольном слове.
3. Напишите программу, заменяющую в произвольном тексте все цифры на символ 'Ж'.
4. Напишите программу, определяющую, имеются ли в произвольном слове два идущих подряд одинаковых символ а.
5. Напишите программу, удаляющую в произвольном слове все буквы, стоящие на четных позициях.
6. Напишите программу, заменяющую в произвольном тексте все строчные латинские гласные на соответствующие заглавные буквы.
7. Напишите программу, подсчитывающую число цифр в произвольном слове.
8. Напишите программу, удваивающую в произвольном слове все буквы, стоящие на нечетных позициях.
9. Напишите программу, шифрующую введенную строку по алгоритму Цезаря со сдвигом $N=3$. При этом шифруемый символ заменяется на другой символ, ASCII-код которого больше на N . Например, 'a' → 'd' 'b' → 'e' и т.д. При вводе используйте ЛАТИНСКИЕ буквы.

Задание 7

Вариант 1,3

Написать программу, которая:

- а) для группы людей создает записи - имя, фамилия, дата рождения (число, месяц, год);
- б) выводит на экран имена и фамилии всех людей этой группы, родившихся в заданном году;
- в) выводит на экран все данные о людях этой группы с заданным именем.

Вариант 2,4

Написать программу, которая:

- а) для группы людей создает записи - имя, фамилия, дата рождения (число, месяц, год);
- а) всех людей из группы, которые родились в январе заданного года;
- б) всех людей из группы, которые родились с 15 марта по 15 апреля (год рождения не важен).

Вариант 5,7

Описать запись со структурой соответствующей каждой задаче. Написать процедуру для ввода вывода записи. Создать массив из 10 записей и решить задачу, используя этот массив.

1. Из ассортимента конфет, выпускаемых кондитерской фабрикой, выбрать те, стоимость которых от 30 до 55 руб. за кг.

Вариант 6,8.

В программе описаны типы:

```

Type Mast = (p, t, b, h); {пики, трефы, бубны,
    червы} Value = (v6, v7, v8, v9, v10, vV, vD,
    vK, vT);
    {шесть,       семь,       восемь,       девять,       десять,       валет,
    дама,
    король, туз}

```

```
Card= record           {описание карты:}  
      m: Mast;         {масть} v:  
      Value{достоинство} end;
```

Написать программу, которая вводит две карты C1 и C2 и козырную масть KM и проверяет, "бьет" ли карта C1 карту C2. Для этого описать логическую функцию Hit (C1, C2: Card; KM: Mast): Boolean

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Жизненный цикл программного обеспечения. Содержание этапов классического ЖЦ. Качество ПО. Критерии качества.
2. Виды архитектур программного средства.
3. Основные области в программировании: методологии и технологии программирования; языки и системы программирования; операционные и архитектурные платформы. Краткая характеристика каждой группы.
4. Методология программирования. Атрибуты методологии. Ядра методологий. Основные методологии: императивного программирования, объектно-ориентированного программирования, функционального программирования, логического программирования, активного программирования (экстремального). Краткая характеристика.
5. Технология: система обозначений, процесс и инструментальные средства.
6. Технологии программирования. Основные понятия и определения.
7. Технологический подход. Классификация подходов.
8. Сложность и сложные системы. Источники сложности. Признаки работоспособной сложной системы. Методы борьбы со сложностью. Моделирование, абстракция, декомпозиция.
9. Функциональная и объектная декомпозиция.
10. Три стратегии конструирования программного обеспечения.
11. Водопадная модель разработки приложений. Недостатки и кризис последовательной модели разработки ПО.
12. Инкрементная модель разработки приложений.
13. Спиральная модель разработки приложений.
14. Тяжеловесные и облегченные процессы разработки приложений.
15. Характеристики XP-процесса.
16. Технологический процесс. Классификация технологических процессов.
17. История и эволюция технологий программирования. Проблемы и перспективы развития.
18. Основные подходы к организации процесса создания и использования программных средств.
19. Методы проектирования программ.
20. Структурное программирование.

21. Модульное программирование. Преимущества модульного программирования. Определение модуля. Свойства модуля. Модульная структура приложения.
22. Сущность объектного подхода к разработке программ.
23. Объектно-ориентированное программирование. Основное понятие ООП. Объекты и классы. Определения, назначение. Принцип инкапсуляции. Принцип наследования. Принцип полиморфизма.
24. Основы визуального программирования. Элементы процесса визуализации. Принципы визуального программирования. Элементы управления. Проектирование формы. Визуальные и объектные аспекты форм. Архитектура приложения. Многооконные интерфейсы.
25. Виды пользовательских интерфейсов и требования к ним.
26. Проектирование пользовательского интерфейса. Рекомендации по проектированию пользовательского интерфейса.
27. Принципы построения пользовательского интерфейса.
28. Три размерности согласованности пользовательского интерфейса.
29. Два вида стилей взаимодействия между пользователем и компьютером и способы для связи.
30. Принципы использования цвета при проектировании эргономичного интерфейса.
31. Тексты и диалоги. Принципы создания текстовых диалогов и отображений.
32. Средства управления графического интерфейса пользователя.
33. Принципы проектирования меню приложения.
34. Режим окон приложения. Три типа окон приложения.
35. Основной элемент интерфейса – экранные формы. Принципы проектирования форм.
36. Проектирование сообщений для пользователя по выполнению необходимых действий на пути решения задачи.
37. Тестирование программ. Методы тестирования.
38. Приемы отладки программных средств. Автономная отладка. Комплексная отладка.
39. Отладка программ. Причины ошибок Типы ошибок в программе: времени выполнения, синтаксические, логические. Защита от ошибок.
40. Диалоговые окна. Стандартные диалоги. Процедуры и функции, реализующие диалоги.
41. Стандарты, необходимые для создания программного обеспечения.
42. Основные программные и эксплуатационные документы (по ГОСТ).
43. Спецификация требований к программной системе. Требования к данным, функциональности, интерфейсу, надежности, безопасности, эффективности. Описание предметной области. Требования к окружению. Архитектура системы.
44. Назначение и содержание «Технического задания».
45. Определение жизненного цикла ПО.
46. Группы процессов, входящие в состав ЖЦ ПО.
47. Наиболее часто используемые процессы в реальных проектах.

48. Определение модели ЖЦ ПО. Используемые модели ЖЦ в настоящее время.
49. Характеристика модели «Водопад».
50. Характеристика спиральной модели.
51. Инструментальные среды разработки и сопровождения ПС. Термин «Case-средства», основные возможности Case-средств.
52. Классификация Case-средств.
53. Структура Case-средства.
54. Сущность прототипной (RAD) технологии. Основные возможности и преимущества быстрой разработки.
55. Классификация инструментальных средств быстрого прототипирования.
56. Основы COM-технологии: понятие интерфейса.
57. Сущность автоматизации OLE.
58. История и эволюция языков моделирования. Блок-схемы, диаграммы.
59. Язык моделирования UML. Возникновение и развитие стандарта. Основные концепции моделирования (абстагирование, множественность представлений). Обзор стандартных элементов и пакетов. Виды диаграмм.
60. Язык моделирования UML. Диаграммы классов. Классы, объекты, отношения ассоциации.
61. Язык моделирования UML. Отношения зависимости, обобщения, реализации. Стереотипы, интерфейсы.
62. Язык моделирования UML. Диаграмма вариантов использования. Применение совместно со спецификацией функций.
63. Язык моделирования UML. Диаграмма развертывания. Статические и динамические диаграммы, как средства изображения структур и процессов.
64. Язык моделирования UML. Диаграмма состояний. Применение при изображении пользовательского интерфейса и протоколов.
65. Язык моделирования UML. Диаграмма деятельности. Аналогии и преимущества перед блок-схемами.
66. Язык моделирования UML. Диаграмма последовательности. Диаграмма кооперации.
67. Компоненты. Каркасы. Паттерны. Паттерны порождения, структуры и управления.
68. Метрики (количественные оценки) ПО. Оценки сложности ПО, качества ПО, эффективности процесса разработки.
69. Правовые аспекты распространения ПО. Copyright и лицензирование. Свободно распространяемое ПО. Аттестация ПС.

3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации

Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий

- оценка **«отлично»** выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90% от общего объёма заданных вопросов;
- оценка **«хорошо»** выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76% от общего объёма заданных вопросов;
- оценка **«удовлетворительно»** выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объёма заданных вопросов;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если количество правильных ответов – менее 60% от общего объёма заданных вопросов.

Критерии формирования оценок по результатам выполнения заданий

«Зачтено»:

- ставится за работу, выполненную полностью без ошибок и недочетов.
- ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.
- ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и двух недочетов.

«Не зачтено» – ставится за работу, если число ошибок и недочетов превысило норму для оценки «удовлетворительно» или правильно выполнено менее 2/3 всей работы.

Виды ошибок:

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*
- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*
- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

Критерии формирования оценок по зачету с оценкой

«Отлично/зачтено» – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний, не допустил логических и фактических ошибок

«Хорошо/зачтено» – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний; допустил незначительные ошибки и неточности.

«Удовлетворительно/зачтено» – студент допустил существенные ошибки.

«Неудовлетворительно/не зачтено» – студент демонстрирует фрагментарные знания изучаемого курса; отсутствуют необходимые умения и навыки, допущены грубые ошибки.

Экспертный лист

оценочных материалов для проведения промежуточной аттестации по дисциплине «Основы и методологии программирования»

по направлению подготовки/специальности

27.03.05 Инноватика
(код и наименование)

Направленность (профиль)/специализация

Управление инновациями
(наименование)

Бакалавр
квалификация выпускника

1. Формальное оценивание			
Показатели		Присутствуют	Отсутствуют
Наличие обязательных структурных элементов:		+	
– титульный лист		+	
– пояснительная записка		+	
– типовые оценочные материалы		+	
– методические материалы, определяющие процедуру и критерии оценивания		+	
Содержательное оценивание			
Показатели	Соответствует	Соответствует частично	Не соответствует
Соответствие требованиям ФГОС ВО к результатам освоения программы	+		
Соответствие требованиям ОПОП ВО к результатам освоения программы	+		
Ориентация на требования к трудовым функциям ПС (при наличии утвержденного ПС)	+		
Соответствует формируемым компетенциям, индикаторам достижения компетенций	+		

Заключение: ФОС рекомендуется/ не рекомендуется к внедрению; обеспечивает/ не обеспечивает объективность и достоверность результатов при проведении оценивания результатов обучения; критерии и показатели оценивания компетенций, шкалы оценивания обеспечивают/ не обеспечивают проведение всесторонней оценки результатов обучения.

Эксперт, должность, ученая степень, ученое звание _____ /

(подпись)