

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Фурсов Владимир Алексеевич

Должность: И.о. директора Пятигорского института (филиал) Северо-Кавказского
федерального университета

Дата подписания: 08.06.2026 13:22:11

Уникальный программный ключ:

1c378726a41fd0143ae5bccc8ba81860b00daa62

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Северо-Кавказский федеральный университет»
Пятигорский институт (филиал) СКФУ
Колледж Пятигорского института (филиал) СКФУ

УТВЕРЖДАЮ

И.О. директора Пятигорского
института
(филиал) СКФУ
В.А. Фурсов

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине	ОП.06 Основы алгоритмизации и программирования
Специальность	09.02.01 Компьютерные системы и комплексы
Форма обучения	очная

Пятигорск 2026

1. Паспорт фонда оценочных средств

1.1. Область применения

Фонд оценочных средств (далее - ФОС) предназначен для оценивания знаний, умений, уровня сформированности компетенций студентов, обучающихся по специальности 09.02.01 Компьютерные системы и комплексы по дисциплине ОП.06 Основы алгоритмизации и программирования.

ФОС составлен на основе ФГОС и рабочей программы дисциплины.

Промежуточная аттестация по дисциплине предусмотрена в форме экзамена с выставлением отметки по системе «отлично, хорошо, удовлетворительно, неудовлетворительно».

1.2. Планируемые результаты освоения дисциплины

ФОС позволяет оценить знания, умения, сформированность общих и профессиональных компетенций в соответствии с требованиями ФГОС и рабочей программой дисциплины.

Планируемые результаты освоения (знания и умения) и перечень осваиваемых компетенций (общих и профессиональных) указываются в соответствии с ФГОС, ОП и рабочей программой дисциплины

умения:

У.1 разрабатывать и анализировать алгоритмы для решения поставленных задач;

У.2 определять сложность алгоритмов;

У.3 реализовывать типовые алгоритмы в виде программ на актуальных языках программирования;

У.4 использовать средства проектирования для создания и графического отображения алгоритмов;

У.5 оформлять код программ в соответствии со стандартом кодирования;

У.6 выполнять проверку, отладку кода программы.

знания:

3.1 понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;

3.2 классификация языков программирования;

3.3 понятие системы программирования;

3.4 основные элементы языка, структура программы;

3.5 методы реализации типовых алгоритмов;

3.6 операторы и операции, управляющие структуры, структуры данных, классы памяти;

3.7 понятие подпрограммы, библиотеки подпрограмм;

3.8 объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения.

общие компетенции:

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.

профессиональные компетенции:

ПК 2.1. Проектировать, разрабатывать и отлаживать программный код модулей управляющих программ.

ПК 2.2. Владеть методами командной разработки программных продуктов.

1.3. Формы контроля и оценивания

Предметом оценки служат умения и знания, предусмотренные ФГОС по дисциплине, направленные на формирование общих и профессиональных компетенций.

Таблица 1 Контроль и оценка освоения дисциплины по темам (разделам)

Элементы дисциплины	Формы контроля и оценивания			
	Текущий контроль		Промежуточная аттестация	
	Методы оценки	Проверяемые ПК, ОК, У, З	Методы оценки	Проверяемые ПК, ОК, У, З
Раздел 1. Основы алгоритмизации				
Тема 1.1. Понятие алгоритма и его свойства	Лабораторное занятие №1, 2 Составление и оформление блок-схем простых алгоритмов.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 З.1, З.2, З.3, З.4, З.5, З.6, З.7, З.8		
Тема 1.2. Методы разработки алгоритмов	Лабораторное занятие №3 Проектирование и оформление алгоритмов сортировки.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 З.1, З.2, З.3, З.4, З.5, З.6, З.7, З.8		
	Лабораторное занятие №4 Проектирование и оформление алгоритмов поиска.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 З.1, З.2, З.3, З.4, З.5, З.6, З.7, З.8		
	Лабораторное занятие №5 Проектирование и оформление сложных алгоритмов.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 З.1, З.2, З.3, З.4, З.5, З.6, З.7, З.8		
Раздел 2. Основы программирования				
Тема 2.1. Базовые понятия программиро	Лабораторное занятие №6 Изучение инструментария	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6		

вания	среды программировани я.	3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №7 Подготовка структуры программы в среде программировани я.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
Тема 2.2. Программна я реализация алгоритмов	Лабораторное занятие №8 Реализация простых циклических алгоритмов.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №9 Реализация алгоритмов обработки одномерных массивов.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №10 Реализация алгоритмов обработки двумерных массивов.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №11 Реализация алгоритмов обработки текстовых данных.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №12 Реализация сложных алгоритмов поиска и ввода- вывода.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
Раздел 3. Основы объектно-ориентированного программирования				
Тема 3.1. Основные	Лабораторное занятие №13, 14	ПК 2.1, ПК 2.2 ОК 01, ОК 02		

понятия объектно- ориентиро- ванного программиро- вания	Создание простейших классов.	У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №15 Создание классов, иерархически связанных между собой.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
Тема 3.2. Реализация методов объектно- ориентиро- ванного программиро- вания	Лабораторное занятие №16 Создание классов для обработки массива данных.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №17 Создание классов для вычисления математических выражений.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
	Лабораторное занятие №18 Разработка проектов с обработкой событий.	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8		
Промежуточная аттестация			Экзамен	ПК 2.1, ПК 2.2 ОК 01, ОК 02 У.1, У.2, У.3, У.4, У.5, У.6 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8

2. Оценочные средства текущего контроля успеваемости и критерии оценки

Комплект заданий для контрольного среза

1-й вариант

1. Понятие алгоритма.
2. Свойства и виды алгоритмов.
3. Критерии «хорошего» алгоритма.

2-й вариант

1. Способы описания алгоритмов: псевдокоды.

2. Линейные алгоритмы.
3. Декомпозиция алгоритма.

Критерии оценивания компетенций

Оценка «5» (отлично) выставляется в случае полного выполнения контрольного среза, отсутствия ошибок, грамотного текста, точность формулировок и т.д.

Оценка «4» (хорошо) выставляется в случае полного выполнения всего объема контрольного среза при наличии несущественных ошибок, не повлиявших на общий результат работы и т.д.

Оценка «3» (удовлетворительно) выставляется в случае недостаточно полного выполнения всех разделов контрольного среза, при наличии ошибок, которые не оказали существенного влияния на окончательный результат, при очень ограниченном объеме используемых понятий и т.д.

Оценка «2» (неудовлетворительно) выставляется в случае, если допущены принципиальные ошибки, контрольный срез выполнен крайне небрежно и т.д.

Фонд тестовых заданий

Вариант 1

1) С какого слова обычно начинается раздел операторов?

- а) uses;
- б) var;
- в) begin;
- г) write.

2) Какая из переменных может не являться целой?

- а) $a:=2$;
- б) $b:=4 \text{ div } 7$;
- в) $c:=-25$;
- г) $d:=d / 6$.

3) Какое значение получит переменная a после выполнения серии команд $a:=5$; $b:=2$; $a:=b * a - \text{SQR}(a)$;

- а) -15
- б) 15
- в) 40
- г) 10

4) Что будет выведено на экран после выполнения серии команд $a:=4$; $b:=8$; $a:=a / b - 5$; $\text{write}(\text{abs}(a))$;

- а) 4.5;
- б) 4.75;
- в) 5.25;
- г) -4.5.

5) Установите соответствие между названиями команд и выполняемыми операциями.

- 1) Write(a);
- 2) Writeln(a);
- 3) Div.

а) после выполнения команды курсор переводится на начало новой строки;

б) команда выполняет целочисленного деления одного числа на другое;

в) после выполнения команды курсор не переводится на начало новой строки.

Вариант 2

1) Установите соответствие между названиями типов данных и их записью в программе.

- 1) INTEGER;
- 2) REAL;
- 3) CHAR;
- 4) STRING.

- а) Строковый тип данных;
- б) Целочисленный тип данных;
- в) Вещественный тип данных;
- г) Символьный тип данных.

2) Установите соответствие между константами и их записью в программе.

- 1) x^2 ;
- 2) $|x|$;
- 3) $\sqrt{x}$.
- а) abs(x);
- б) sqrt(x);
- в) sqr(x).

3) Установите правильную последовательность записи цикла с предусловием:

- а) Do <тело цикла>;
- б) Begin;
- в) End;
- г) While <выражение>.

4) Установите правильную последовательность записи цикла с постусловием:

- а) Begin
- б) Until <условие>;
- в) Repeat <тело цикла>;
- г) End.

5) Установите правильную последовательность записи цикла с параметром:

- а) To <конечное значение>;

- б) Do <тело цикла>;
- в) For<счетчик>:=< начальное значение> ;
- г) Begin;
- д) End.

3. Оценочные средства для промежуточной аттестации и критерии оценки

Вопросы к экзамену

1. Понятие алгоритма.
2. Свойства и виды алгоритмов.
3. Способы описания алгоритмов: псевдокоды.
4. Правила составления блок-схем.
5. Стандарты графического оформления алгоритмов.
6. Линейные алгоритмы.
7. Разветвляющиеся алгоритмы.
8. Циклические алгоритмы.
9. Критерии «хорошего» алгоритма.
10. Основные методы и этапы проектирования алгоритмов.
11. Нисходящее и восходящее проектирование.
12. Структурное и модульное программирование
13. Алгоритмы поиска.
14. Алгоритмы сортировки.
15. Вложенные циклы.
16. Вспомогательные алгоритмы.
17. Различные комбинации алгоритмических конструкций.
18. Алгоритм Евклида.
19. Декомпозиция алгоритма.
20. Классификация языков программирования.
21. Понятие системы программирования.
22. Основные элементы языка.
23. Структура типовой программы.
24. Особенности среды программирования.
25. Переменные.
26. Типы данных.
27. Объявление и инициализация переменных.
28. Область действия и время существования переменных.
29. Константы.
30. Операторы и операции.
31. Ввод – вывод данных.
32. Операторы присваивания.
33. Операторы отношения.

34. Оператор выбора.
35. Операторы перехода.
36. Операторы цикла.
37. Принудительный выход из цикла.
38. Массивы: определение, виды.
39. Объявление одномерного массива.
40. Ввод и вывод одномерных массивов.
41. Обработка одномерных и двумерных массивов.
42. Управляющие структуры.
43. Понятие потока.
44. Механизм буферизации.
45. Понятие подпрограммы.
46. Библиотеки среды разработки
47. Понятие класса и объекта.
48. Наследование.
49. Полиморфизм.
50. Инкапсуляция.
51. Метод: понятие, правила записи.
52. Инкапсуляция как управление доступом к данным.
53. Свойства класса: понятие, виды, правила записи.
54. Иерархия классов.
55. Интерфейсы.
56. Переменные ссылочного типа.
57. Конструкторы.
58. Синтаксис наследования.
59. Обработка события.

Критерии оценивания компетенций

Оценка «отлично» выставляется студенту, глубоко и прочно усвоившему программный, в том числе лекционный материал, последовательно, четко и самостоятельно (без наводящих вопросов) отвечающему на вопрос билета.

Оценка «хорошо» выставляется студенту, твердо знающему программный, в том числе лекционный материал, грамотно и по существу, отвечающему на вопрос билета и не допускающему при этом существенных неточностей (неточностей, которые не могут быть исправлены наводящими вопросами или не имеют важного практического значения). То же относится к освещению практически важных вопросов.

Оценка «удовлетворительно» выставляется студенту, который обнаруживает знание основного материала, но не знает его деталей, допускает неточности, недостаточно правильные формулировки, излагает материал с нарушением последовательности, отвечает на практически важные вопросы с помощью или поправками экзаменатора.

Оценка «неудовлетворительно» выставляется студенту, который не знает

значительной части программного, в том числе лекционного материала.

Таблица 2 – Ключи к вопросам фонда оценочных средств

№	Компетенция	Содержание вопроса	Правильный ответ
Контрольный срез Вариант 1			
1.	ПК 2.1	Понятие алгоритма.	Алгоритм – это конечный набор правил, который определяет последовательность операций для решения конкретного множества задач.
2.	ПК 2.1	Свойства и виды алгоритмов.	Алгоритм должен обладать следующими свойствами: дискретностью; массовостью; определённостью; результативностью; формальностью. Различают следующие виды алгоритмов: линейный; разветвляющийся; циклический.
3.	ПК 2.1	Критерии «хорошего» алгоритма.	Есть несколько критериев для оценки хорошего алгоритма: Корректность; Удобочитаемость; Надежность; Обобщенность; Эффективность.
Контрольный срез Вариант 2			
4.	ПК 2.1	Способы описания алгоритмов: псевдокоды.	К настоящему времени сложились пять наиболее употребительных способов записи: словесный,

			формульно-словесный, графический, при помощи псевдокодов и языков программирования.
5.	ПК 2.1	Линейные алгоритмы.	Линейный алгоритм - это алгоритм, образуемый командами, которые выполняются однократно и именно в той последовательности, в которой записаны.
6.	ПК 2.1	Декомпозиция алгоритма.	Под декомпозицией алгоритма понимают разложение его общей алгоритмической схемы на вспомогательные алгоритмы и головной алгоритм.
Фонд тестовых заданий Вариант 1			
7.	ПК 2.1	1) С какого слова обычно начинается раздел операторов? а) uses; б) var; в) begin; г) write.	в
8.	ПК 2.1	2) Какая из переменных может не являться целой? а) a:=2; б) b:=4 div 7; в) c:=-25; г) d:=d / 6.	г
9.	ПК 2.1	3) Какое значение получит переменная a после выполнения серии команд a:=5; b:=2; a:=b * a - SQR(a) ; а) -15	а

		б) 15 в) 40 г) 10	
10.	ПК 2.1	4) Что будет выведено на экран после выполнения серии команд a:=4; b:=8; a:=a / b - 5; write(abs(a)); а) 4.5; б) 4.75; в) 5.25; г) -4.5.	а
11.	ПК 2.1	5) Установите соответствие между названиями команд и выполняемыми операциями. 1) Write(a); 2) Writeln(a); 3) Div. а) после выполнения команды курсор переводится на начало новой строки; б) команда выполняет целочисленного деления одного числа на другое; в) после выполнения команды курсор не переводится на начало новой строки.	1. в 2. а 3. б
Вариант 2			
12.	ПК 2.1	1) Установите соответствие между названиями типов данных и их записью в программе. 1) INTEGER; 2) REAL; 3) CHAR; 4) STRING. а) Строковый тип данных; б) Целочисленный тип данных; в) Вещественный тип данных; г) Символьный тип данных.	1. б 2. в 3. г 4. а
13.	ПК 2.1	2) Установите соответствие между константами и их записью в программе. 1) x^2 ; 2) $ x $;	1. в 2. а 3. б

		3) $\sqrt{x}$. а) abs(x); б) sqrt(x); в) sqr(x).	
14.	ПК 2.1	3) Установите правильную последовательность записи цикла с предусловием: а) Do <тело цикла>; б) Begin; в) End; г) While <выражение>.	б,г,а,в
15.	ПК 2.1	4) Установите правильную последовательность записи цикла с постусловием: а) Begin б) Until <условие>; в) Repeat <тело цикла>; г) End.	а,в,б,г
16.	ПК 2.1	5) Установите правильную последовательность записи цикла с параметром: а) To <конечное значение>; б) Do <тело цикла>; в) For<счетчик>:=< начальное значение> ; г) Begin; д) End.	г,в,а,б,д