

Федеральное государственное образовательное бюджетное
учреждение высшего образования
**«Финансовый университет при Правительстве Российской Федерации»
(Финуниверситет)**

**Самарский финансово-экономический колледж
(Самарский филиал Финуниверситета)**

УТВЕРЖДАЮ
Заместитель директора по учебно-
методической работе _____ Л.А Косенкова
« 21 » февраля 20 22 г.



**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ «ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И
ПРОГРАММИРОВАНИЯ»
СПЕЦИАЛЬНОСТЬ: 09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И
ПРОГРАММИРОВАНИЕ**

Фонд оценочных средств по дисциплине разработан в соответствии с рабочей программой по дисциплине «Основы алгоритмизации и программирования», с федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки Российской Федерации от 09.12.2016 года № 1547

Присваиваемая квалификация: администратор баз данных

Разработчики:

Платковская Е.А.

Преподаватель Самарского филиала
Финуниверситета

Чурилов А.С.

Преподаватель Самарского
филиала Финуниверситета

Рецензент:

Шарамыгина Т.В.

Директор ООО «Ризотек»



Фонд оценочных средств дисциплины рассмотрен и рекомендован к утверждению на заседании предметной (цикловой) комиссии естественно-математических дисциплин

Протокол от « 24 » января 20 22 г. № 5

Председатель ПЦК _____ М.В. Писцова

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Фонд оценочных средств предназначен для контроля и оценки уровня результатов освоения необходимых знаний и умений обучающихся, осваивающих программу учебной дисциплины ОП.04. Основы алгоритмизации и программирования специальности среднего профессионального образования 09.02.07 Информационные системы и программирование.

В результате освоения учебной дисциплины обучающийся должен **уметь**:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.

В результате освоения учебной дисциплины обучающийся должен **знать**:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм.
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.

В процессе практических занятий обеспечивается формирование и развитие универсальных учебных действий в контексте преемственности формирования общих компетенций и профессиональных компетенций:

Код	Наименование компетенций
ОК 01.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
ОК 02.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 04.	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
ОК 09.	Использовать информационные технологии в профессиональной деятельности.
ОК 10.	Пользоваться профессиональной документацией на государственном и иностранном языке.
ПК 1.1.	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
ПК 1.2.	Разрабатывать программные модули в соответствии с техническим заданием.
ПК 1.3.	Выполнять отладку программных модулей с использованием специализированных программных средств.
ПК 1.4.	Выполнять тестирование программных модулей.

ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода.
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.
ПК 4.1	Осуществлять инсталляцию, настройку и обслуживание программного обеспечения компьютерных систем.
ПК 4.2	Осуществлять измерения эксплуатационных характеристик программного обеспечения компьютерных систем
ПК 7.1.	Выявлять технические проблемы, возникающие в процессе эксплуатации баз данных и серверов.
ПК 7.2.	Осуществлять администрирование отдельных компонент серверов.
ПК 7.3.	Формировать требования к конфигурации локальных компьютерных сетей и серверного оборудования, необходимые для работы баз данных и серверов.
ПК 7.4.	Осуществлять администрирование баз данных в рамках своей компетенции.
ПК 7.5.	Проводить аудит систем безопасности баз данных и серверов с использованием регламентов по защите информации.

**ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ**

Результаты обучения (освоенные умения, усвоенные знания)	ОК	Наименование темы	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
<p>Уметь:</p> <ul style="list-style-type: none"> – Разрабатывать алгоритмы для конкретных задач. – Использовать программы для графического отображения алгоритмов. – Определять сложность работы алгоритмов. – Работать в среде программирования. – Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. <p>Знать:</p> <ul style="list-style-type: none"> – Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. – Эволюцию языков программирования, их классификацию, понятие системы программирования. 	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5	Тема 1.1. Языки программирования	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5	Тема 1.2. Типы данных	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена

<p>– Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.</p> <p>– Подпрограммы, составление библиотек подпрограмм</p> <p>– Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</p>	<p>ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5</p>	<p>Тема 2.1. Операторы языка программирования</p>	<p>Вопросы для устного и письменного опроса по теме:</p> <p>Практическое занятие выполнение заданий по теме</p>	<p>Вопросы для проведения экзамена</p>
	<p>ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5</p>	<p>Тема 3.1. Процедуры и функции</p>	<p>Вопросы для устного и письменного опроса по теме:</p> <p>Практическое занятие выполнение заданий по теме</p>	<p>Вопросы для проведения экзамена</p>
	<p>ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3</p>	<p>Тема 3.2. Структуризация в программировании</p>	<p>Вопросы для устного и письменного опроса по теме:</p> <p>Практическое занятие выполнение заданий по теме</p>	<p>Вопросы для проведения экзамена</p>

	ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5			
	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5	Тема 3.3. Модульное программирование	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3	Тема 4.1 Указатели	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена

ПК 7.4 ПК 7.5				
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5		Тема 5.1 Основные принципы объектно- ориентированного программирования (ООП)	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для про- ведения экзамена
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5		Тема 5.2 Интегрированная среда разработчика	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для про- ведения экзамена
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1		Тема 5.3. Визуальное событийно - управляемое программирование	Вопросы для устного и письменного опроса по теме: Практическое занятие	Вопросы для про- ведения экзамена

	ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5		выполнение заданий по теме	
	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5	Тема 5.4 Разработка оконного приложения	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 01 ОК02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2	Тема 5.5 Этапы разработки приложений	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена

	ПК 7.3 ПК 7.4 ПК 7.5			
	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ОК 10 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 4.1 ПК 2.4 ПК 2.5 ПК 4.2 ПК 7.1 ПК 7.2 ПК 7.3 ПК 7.4 ПК 7.5	Тема 5.6 Иерархия классов	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для про- ведения экзамена

Материал для текущего контроля

Тема 1.1. Языки программирования

1. Переменная — это величина, которая имеет ...

1. Значение
2. имя
3. тип
4. все вышеперечисленное +

2. Оператор «ввод» в Паскале является процедурой ...

1. write +
2. input
3. read

3. Объявляются ли переменные в языке Python?

1. Да
2. Нет +

4. Операторы размещают между словами ...

1. нач и кон +
2. старт и финиш

5. Как называется алгоритм, который целиком используется в составе другого алгоритма?

1. циклический
2. вспомогательный +
3. служебный

6. Что такое структурное программирование?

1. Технология разработки программного обеспечения, основой которой служит представление программы в виде логарифмических уравнений
2. Технология разработки программного обеспечения, основой которой служит представление программы в виде математических формул и символов естественного языка
3. Технология разработки программного обеспечения, основой которой служит представление программы в виде иерархической конструкции логических целостных фрагментов +

7. Оператор — это ...

1. значение
2. тело программы
3. команда +

8. В языке Паскаль конец обозначается как ...

1. end
2. end с точкой на конце +
3. fin

9. В языке Паскаль переменные предваряют словом ...

1. не объявляют
2. var +
3. ver

10. Оператор присваивания содержит символы ...

1. «:»»
2. «:=» +
3. «?!»

11. Программа начинается ключевым словом «алг», если она ...

1. на студенческом алгоритмическом языке
2. на школьном алгоритмическом языке +
3. на любом алгоритмическом языке

12. Как называется алгоритм, который на каком-либо шаге обращается сам к себе?

1. циклическим
2. рекурсивным +
3. вспомогательным

Тема 1.2. Типы данных

1. Раздел var это ...

1. Раздел имя программы
2. Раздел описания переменных +
3. Начало программы
4. Конец программы
5. Раздел подключения библиотек

2. Тип integer предназначен для переменных в которых хранятся ...

1. Символы
2. Любые числа
3. Действительные числа
4. Целые числа +

3. Какого пункта нет в структуре программы?

1. Название программы
2. Раздел пояснения программы +
3. Тело программы
4. Раздел описания переменных

4. После какой команды мы описываем переменные?

1. uses
2. program
3. begin
4. end
5. var +

5. Выражение a/b будет принадлежать к ...

1. типу string
2. типу real +
3. типу char
4. типу integer

6. В рабочем окне системы программирования Pascal есть:

1. Панель инструментов +
2. Строка заголовка окна +
3. Рабочая область экрана +

4. Строка меню +
5. Строка прокрутки

7. Каждое выражение (каждый оператор) в программе отделяется друг от друга ...

1. Точкой
2. тире
3. дефисом
4. точкой с запятой +
5. запятой

8. Какие из команд при выполнении программы выведут в окне вывода надпись: Привет!!!

1. Написать('Привет!!!');
2. Write('Привет!!!'); +
3. Write('Привет это я!!!');
4. Write('Привет')
5. read(Привет!!!);

9. В каком из типов переменных мы можем хранить не только числа но и символы?

1. real
2. string +
3. integer

10. Какое из чисел является целым?

1. 2,3
2. 4/7
3. -1 +

Тема 2.1. Операторы языка программирования

1. При каком исходном значении переменной X результатом выполнения команды $X:=X \bmod 3$ будет 0?

1. 5
2. 4
3. при любом, кратном 3; +
4. 2

2. Сколько существует видов разветвленного алгоритма?

1. 5
2. 8
3. 2 +
4. 3

3. Сколько вариантов действий можно осуществить при наличии одного разветвления в программе?

1. 5
2. 2
3. 1 +

4. Когда вы используете выражение, в котором есть слово then, то в выражении обязательно должно быть слово:

1. writeln.
2. if; +

3. else;
4. readln;

**5. Какое значение будет иметь переменная b при значении a := -20, в примере:
if a >= 15 then b:= a * 2 else b:= a — 10?**

1. -29
2. 30
3. -30 +

6. real - это тип переменных

1. вещественных +
2. целых
3. логических

7. Оператор вывода (печати) - это

1. real
2. write +
3. var

8. var - это

1. раздел описания меток
2. раздел описания типов
3. раздел описания переменных +

9. Оператор присваивания правильно записан

a:=

a:=

a=

1. 1
2. 2 +
3. 3

10. Форма организации действий, при которой одно и то же действие выполняется несколько раз до тех пор, пока соблюдается некоторое условие:

1. циклический алгоритм +
2. линейный алгоритм
3. ветвление

Тема 3.1. Процедуры и функции

1. Программа, предназначенная для решения вывода на экран результата работы конкретной частной задачи и к которой обращаются из другой программы называется...

1. модуль
2. функция
3. подпрограмма
4. процедура +

2. Заголовок процедуры начинается со слова ...

1. procedure +
2. function
3. program

4. нет правильного ответа

3. Процедура может возвращать в основную программу...

1. единственное значение
2. несколько значений
3. 0 значений
4. все варианты правильные +

4. В приведенных фрагментах описания заголовков функций выберите правильные:

1. Function Sum1 (a,b:Integer; Var c:integer): real; +
2. function Sum (a: integer; Var c: integer);
3. Function 2S (a, b): integer;
4. Function SQ (a,c: real): integer; +

5. В приведенных фрагментах описания заголовков процедур выберите ошибочные:

1. Procedure P (Var a : Integer);
2. Procedure P (a : Integer);
3. Procedure P (Var a, b : Integer);
4. Procedure P (a, b); +

6. Каковы отличия процедуры от функции?

Ответ: _____

7. Подпрограмма, которая всегда возвращает какое-либо значение это?

1. Функция +
2. Процедура

8. Как на русском языке называется тип данных String?

1. Целочисленный
2. Вещественный
3. Строковый +
4. Символьный

9. Какая из функций переводит числовое значение в строковое?

1. str() +
2. insert()
3. val()
4. pos()

10. У нас есть строковая переменная со значением s="паровоз". В новую переменную s2 запишем функцию Copy в таком виде: s2:=Copy(s,3,3). Какая из приведенных строк будет сохранена в переменную s2?

1. пар
2. ров +
3. воз
4. ово

Тема 3.2. Структуризация в программировании

1. Из каких основных элементов состоит программа на языке Паскаль?

1. Заголовок программы, блок описания используемых данных и программный блок +

2. Заголовок программы, блок описания действий
3. Заголовок программы, блок управления и блок помощи

2. Каким знаком разделяются переменные разного типа?

1. двоеточием
2. точка с запятой +
3. запятой

3. Какой знак используется для разделения переменных одного типа?

1. точка
2. запятая +
3. точка с запятой

4. Как называются языковые конструкции, при помощи которых в программах записываются действия, которые выполняются в процессе решения задачи над данными?

1. операндами
2. операторами +
3. операториями

5. С помощью чего осуществляется присваивание переменной нового значения?

1. С помощью оператора присваивания +
2. С помощью операнда присваивания
3. С помощью операнда присвоения

6. После какого знака указывается тип переменных одного типа?

1. После дефиса
2. После точки с запятой
3. После двоеточия +

7. Что такое паскаль?

1. Универсальный язык программирования +
2. Главный язык программирования
3. Аспектно-ориентированный язык программирования

8. В чью честь был назван язык Паскаль?

1. В честь математика Блеза Паскаля +
2. В честь Петра Паскаля
3. В честь Пьера Паскаля

9. Каким словом начинается и заканчивается программный блок?

1. Начинается со слова begin, заканчивается словом finish
2. Начинается со слова start, заканчивается словом end с точкой
3. Начинается со слова begin, заканчивается словом end с точкой +

10. Какой знак нужно ставить перед оператором end?

1. Точку
2. Точку с запятой
3. Знак не нужен +

Тема 3.3. Модульное программирование

1. Будущее окно приложения, на котором будут располагаться компоненты, называется:

1. Редактор
2. Формой +
3. Приложение
4. Модулем

2. Размер модуля измеряется числом содержащихся в нем:

1. Строк +
2. Линий
3. Переменных

3. Технология, основанная на представлении программ в виде совокупности объектов, каждый из которых является реализацией собственного класса, которые в свою очередь образуют иерархию на принципах наследования:

1. Структурное программирование
2. Модульное программирование
3. Объектно-ориентированное программирование +

4. Свойство Name

1. Текст, который отображается на элементе или вблизи элемента
2. Строка текста, которая представляет фактические данные, которые этот объект содержит
3. Идентификатор, под которым элемент управления определен в программе +

5. Действие, распознаваемое объектом

1. Свойство
2. Событие +
3. Метод

6. Худшим видом сцепления модулей является:

1. По внешним ссылкам
2. По данным
3. По содержимому +

7. Текстовая часть программы пишется непосредственно в окне:

1. Редактора кода +
2. Инспектора объектов
3. Палитры компонентов
4. Проектировщика формирования

8. Внешняя часть класса

Общими для большинства компонентов являются свойства:

1. Цвет +
2. Имя +
3. Размер +
4. Интервал
5. Положение на экране.

9. Будущее окно приложения, на котором будут располагаться компоненты, называется:

1. Редактор

2. Формой +
3. Приложение
4. Модулем

10. Разработка алгоритма решения задачи — это

1. Сведение задачи к математической модели, для которой известен метод решения
2. Выбор наилучшего метода из имеющихся
3. Точное описание данных, условий задачи и ее целого решения
4. Определение последовательности действий, ведущих к получению результатов +

Тема 4.1 Указатели

1. Что представляют собой показатели?

1. Селекторы
2. Спецификаторы
3. Переменные +

2. Значениями указателей являются:

1. Адреса объектов +
2. Переменные
3. Идентификаторы

3. Если адрес объекта помещен в указатель, то доступ к объекту является:

1. Косвенным +
2. Опосредованным
3. Детализированным

4. Автоматическое уменьшение на 1 носит название:

1. Детерминирование
2. Декрементирование +
3. Модулирование

5. Инкрементирование — это

1. Увеличение на единицу +
2. Объявление переменных
3. Вывод данных

6. При объявлении типизированных указателей указывается:

1. Тип данных +
2. Тип связей и ссылок
3. Тип соответствий имен входы

7. Указатели, которые могут «смотреть» на данные любого типа, носят название:

1. Структурные
2. Априорные
3. Нетипизированные +

8. Перед именем указателя при его объявлении ставится:

1. @
2. * +
3. {

9. Инициализацию указателя можно совместить:

1. С его идентификацией
2. С его объявлением +
3. С его контекстным выводом

10. Значение указателя можно задать с помощью:

1. Оператора детерминации
2. Статистического оператора возврата значения
3. Оператора присваивания +

Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)

1. Один из принципов объектно-ориентированного программирования:

1. Инкапсуляция +
2. Ингаляция
3. Инструкция

2. Данные, характеризующие состояние объекта:

1. Доли объекта
2. Части объекта
3. Атрибуты объекта +

3. Под объектами понимают:

1. Всю абстрактную сущность, заданную набором имен атрибутов и имен методов поведения
2. Некоторую абстрактную сущность, заданную набором имен атрибутов и имен методов поведения +
3. Некоторую видимую сущность, заданную набором имен атрибутов и имен методов поведения

4. Изменение состояния объекта в ответ на какое-либо действие:

1. Значение
2. Событие +
3. Данность

5. Действие, которое может выполнить объект:

1. Метод +
2. Событие
3. Свойство

6. Компоненты, которые видны во время работы приложения, с ними напрямую может взаимодействовать пользователь, называются:

1. Абстрактными
2. Видимыми
3. Визуальными +

7. Свойство Align отвечает за:

1. Уменьшение компонента на форме
2. Увеличение компонента на форме
3. Выравнивание компонента на форме

8. Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам:

1. Статическая
2. Дружеская +
3. Шаблонная

9. Имя формы, используется для управления формой и доступа к компонентам формы:

1. Свойство формы Name +
2. Значение формы Name
3. Следствие формы Name

10. Этот компонент предназначен для вывода текста на поверхность формы:

1. Label +
2. Edit
3. Button

Тема 5.2 Интегрированная среда разработчика

1. Что такое IDE?

1. Формат ввода данных
2. Интегрированная среда разработки +
3. Шаблон данных

2. Интегрированная среда разработки впервые появилась в системе:

1. Turbo Pascal +
2. Delphi
3. Visual Studio

3. Какой режим загрузки ЭВМ предшествовал интегрированной среде разработки?

1. Модульный
2. Пакетный +
3. Конструктивный

4. Основным режимом работы на персональном компьютере является:

1. Диалоговый +
2. Пакетный
3. Контекстный

5. К составляющим частям интегрированной среды разработки следует отнести:

1. Отладчик +
2. Загрузчик +
3. Редактор связей +

6. Сохранение файла производится нажатием клавиши:

1. F2 +
2. F6
3. F12

7. К исключительным ситуациям следует отнести:

1. Деление на нуль +

2. Переполнение +
3. Выход данных из шаблона

8. По команде Contents на экране появляются названия:

1. Методов и классов
2. Шаблонов данных
3. Справочных разделов +

9. Для устранения синтаксических ошибок используется:

1. Модулятор
2. Компилятор +
3. Деструктор

10. Результатом работы компилятора является:

1. Объектный модуль +
2. Контекстное приложение
3. Идентификаторы ввода

Тема 5.3 Визуальное событийно-управляемое программирование

1. Системы программирования, позволяющие создавать программы с оконным интерфейсом называют __ приложениями (**оконными**)

2. Основателем RAD-технологии считается:

1. Роберт Флойд
2. Джеймс Мартин +
3. Скотт Бойм

3. Технология создания программ, основанная на использовании системы объектов, называется:

1. Инспектором объектов
2. Обработчиком событий
3. Объектно-ориентированным программированием +

4. Внешний вид окна будущего приложения строится на:

1. Платформе
2. Объекте
3. Форме +

5. Обработчик событий — это:

1. Метод программирования, основанный на управлении событиями
2. Процедура (или функция) иницируемая событием +
3. Совокупность данных и методов работы с ними

6. Объектно-событийная модель программы предполагает следующее:

1. Создание оконного приложения
2. Создание объектов с присущими им свойствами и методами +
3. Организация данных внутри объектов

7. Палитра компонентов содержит:

1. Список визуальных компонентов, объединенных в группы +
2. Инспектор объектов

3. Контейнеры

8. Обращение к свойствам и методам объекта в языке VBA имеет следующий синтаксис:

1. Объект.Свойство +
2. Действие. Объект
3. Объект. Метод. Свойство

Тема 5.4 Разработка оконного приложения

1. Во сколько этапов осуществляется разработка оконного приложения:

1. Два
2. Три +
3. Четыре

2. Средство взаимодействия пользователя с программой:

1. Интерфейс +
2. Приложение
3. Внешнее устройство

3. На этом этапе определяют, какая информация будет выводиться на экран:

1. Создание интерфейса
2. Разработка сценария +
3. Разработка алгоритма

4. Массив — это...

1. это набор данных, следующих друг за другом +
2. Возникшая в процессе выполнения программы ошибка
3. Реакция приложения на операцию ввода
4. Используемая в процессе написания программы инструкция, для обозначения комментария

5. Объектно-ориентированное программирование — это

1. применение знания языка программ. Pascal для программирования в среде Delphi
2. методика разработки программ, в основе которой лежит понятие объектам +
3. формирование научного мировоззрения, привитие интереса к предмету
4. процесс конструирования визуального проявления программы

6. Что является основой программы?

1. метка
2. компонент
3. форма +
4. формат

7. Программный объект включает в себя...

1. несколько модулей
2. несколько процедур по обработке данных
3. данные и процедуры по обработке этих данных +
4. форму и расположенные на ней элементы управления

Тема 5.5 Этапы разработки приложений

1. Первый этап в жизненном цикле программы:

1. Формулирование требований +
2. Проектирование
3. Автономное тестирование

2. В стадии разработки программы не входит:

1. Постановка задачи
2. Автоматизация программирования
3. Составление спецификаций

3. Самый важный критерий качества программы:

1. Надежность
2. Работоспособность +
3. Быстродействие

4. Один из необязательных этапов жизненного цикла программы:

1. Тестирование
2. Анализ требования
3. Оптимизация +

5. Какой этап выполняется раньше:

1. Отладка
2. Оптимизация
3. Программирование +
4. Тестирование

6. Самый большой этап в жизненном цикле программы:

1. Программирование
2. Эксплуатация +
3. Тестирование

7. Что относится к этапу программирования:

1. Написание кода программы +
2. Разработка интерфейса
3. Анализ требований

8. Последовательность этапов программирования:

1. Компилирование, компоновка, отладка +
2. Отладка, компоновка, компилирование
3. Компилирование, отладка, компоновка

9. Что выполняется раньше:

1. Написание исходного кода
2. Разработка алгоритма +
3. Компиляция

10. На каком этапе производится выбор языка программирования:

1. Отладка
2. Тестирование
3. Проектирование +

Тема 5.6. Иерархия классов

1. Иерархия классов — это:

1. Структура многочисленных связанных классов, определяющая, какие классы наследуют функции от других классов +
2. Совокупность классов, которые передают функции других классов
3. Классы, располагающиеся от высшего к низшему

2. Новые классы, формируемые на основе базовых, называют:

1. Родственными
2. Производными +
3. Непроизводными

3. Сколько класс имеет предков:

1. Один
2. Два и более
3. Не имеет

4. Укажите верный синтаксис иерархии:

1. [атрибуты] [спецификаторы] class имя_класса [: предки] { тело_класса } +
2. [спецификаторы] [атрибуты] class имя_класса [: предки] { тело_класса }
3. class имя_класса [: предки] [спецификаторы] [атрибуты] { тело_класса }

5. Наследование — это:

1. Механизм сокрытия классов
2. Механизм преобразования классов на основе уже существующего
3. Механизм создания нового класса на основе уже существующего +

6. Какое основное значение механизма наследования?

1. Обеспечение создания новых классов
2. Повторное использование кодов +
3. Скрыть классы

7. Какое наследование скрывает классы и элементы классов:

1. Частное наследование +
2. Полное наследование
3. Частичное наследование

8. В каком порядке выполняются конструкторы:

1. В порядке наследования +
2. В обратном порядке
3. В любом

9. Что наследует производный класс:

1. Другой класс
2. Функции
3. Переменные +

10. В каком порядке выполняются деструкторы:

1. В порядке наследования

2. В обратном порядке +
3. В любом

**Фонд оценочных средств промежуточной аттестации
по дисциплине ОП.04. «Основы алгоритмизации и программирования».**

Промежуточная аттестация обучающихся:

- обеспечивает оперативное управление учебной деятельностью обучающегося и её корректировку;
- проводится с целью определения соответствия уровня и качества подготовки обучающегося требованиям к результатам ОПОП.

Промежуточная аттестация обучающихся проводится в форме устного экзамена дисциплине ОП.04. Основы алгоритмизации и программирования.

Материалы для промежуточной аттестации составлены на основе рабочей программы дисциплины и охватывают ее наиболее актуальные разделы и темы.

На основе разработанного и объявленного студентам перечня вопросов, практических заданий, рекомендуемых для подготовки к экзамену, составляются билеты для проведения экзамена. Содержание билетов для экзамена в том числе практических заданий, до сведения студентов не доводится.

Оценка качества подготовки студентов осуществляется в двух основных направлениях:

- оценка уровня освоения дисциплины;
- оценка компетенций обучающихся.

В критерии оценки уровня подготовки студента входят:

- уровень освоения студентом материала, предусмотренного учебной программой по дисциплине;
- умение студента использовать теоретические знания при выполнении практических заданий;
- обоснованность, четкость, краткость изложения ответа.

Уровень подготовки студента оценивается в баллах: 5 (отлично), 4 (хорошо), 3 (удовлетворительно), 2 (неудовлетворительно).

Оценкой «отлично» оцениваются глубокое и полное понимание программного материала, умение самостоятельно разъяснять изученные положения, убедительность и ясность ответа, когда студент не допускает ошибок, логически и литературно правильно излагает материал, свободно ориентируется в действующем законодательном и инструктивном материале, а также при условии выполнения практических заданий на 100 – 90% (если они решены методически и арифметически верно).

Оценкой «хорошо» оценивается глубокое и правильное усвоение программного материала, однако в ответе допускаются неточности и незначительные ошибки как в содержании, так и в форме построения ответа, а также при условии выполнения практических заданий на 90-80% (если они выполнены методически верно, но имеют место незначительные арифметические ошибки).

Оценка «удовлетворительно» свидетельствует о том, что студент знает основные существенные положения учебного материала, но не умеет их четко разъяснить, допускает отдельные ошибки и неточности в содержании материала и в форме построения ответа, кроме того, практические задания экзаменационного билета выполнены на 80-60% при наличии незначительных ошибок в методике расчетов (которые, однако, искажают итоговый результат работы).

Оценка «неудовлетворительно» выставляется в том случае, если студент знаком с учебным материалом, но не отражает в нем системы знаний, не выделяет основные положения, допускает существенные ошибки, которые искажают смысл полученного, а практические задания выполнены с существенными ошибками в методике расчетов; либо при неполном (менее 60%) решении практических заданий, не дающем представления о системности знаний студента по данному вопросу.

Кроме того, оценка «неудовлетворительно» выставляется в том случае, если студент применяет на экзамене неразрешенные к использованию материалы.

Перечень экзаменационных вопросов
по дисциплине **ОП.04. Основы алгоритмизации и программирования**
Перечень вопросов для подготовки к экзамену

1. Перечислите составляющие единицы алфавитов языков программирования. Опишите структуру программы на языке программирования Паскаль.
2. Опишите структуру программы на языке программирования. Приведите примеры директив препроцессора языка программирования, поясните их назначение и порядок работы в программе.
3. Дайте определение понятия «тип данных». На какие группы делятся все типы данных? Перечислите типы данных относящиеся к группе простых типов. Перечислите простые типы данных языка программирования Паскаль.
4. Дайте определение понятия «тип данных». На какие группы делятся все типы данных? Перечислите типы данных относящиеся к группе простых типов. Перечислите простые типы данных языка программирования.
5. Дайте определение понятия «тип данных». Опишите правила построения символьных данных в языках программирования Паскаль.
6. Дайте определение понятия «тип данных». Опишите порядок использования вещественных типов данных в языках программирования Паскаль.
7. Дайте определение понятия «тип данных». Опишите использование логических данных в языках программирования Паскаль.
8. Опишите порядок использования оператора присваивания и правила построения выражений в языках Паскаль. Назовите операции над данными простых типов в языках Паскаль.
9. Расскажите, как построен ввод и вывод данных в языке программирования Паскаль.
10. Расскажите, как построен ввод и вывод в языке программирования.
11. Опишите правила реализации конструкции ветвления в языке программирования Паскаль.
12. Опишите правила реализации конструкция ветвление в языке программирования
13. Опишите правила реализации циклов с неизвестным числом повторов в языке программирования Паскаль.
14. Опишите правила реализации циклов с неизвестным числом повторов в языке программирования.
15. Опишите правила реализации циклов с параметром в языке программирования Паскаль.
16. Опишите правила реализации циклов с параметром в языке программирования.
17. Дайте определение структуры данных - массив. Опишите правила реализации массивов в языке Паскаль.
18. Дайте определение структуры данных - массив. Опишите правила реализации массивов в языке.
19. Дайте определение структуры данных строка. Опишите правила реализации строк.
20. Дайте определение структуры данных файл. Опишите правила реализации файлов.
21. Приведите примеры визуальных компонент среды Delphi. Поясните порядок работы с ними.

Примерные задания

1. Решите предложенную задачу. Реализуйте решение в виде программы на языке программирования Паскаль. Подтвердите правильность решения тестированием.
2. Дано число k ($0 < k < 11$) и таблица размера 4×10 заполненная целыми числами. Найти сумму и произведение элементов k -го столбца данной таблицы.
3. Дана таблица размера 5×9 заполненная целыми числами. Найти суммы элементов в каждой четной строке и каждом четном столбце.
4. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное и максимальное значение в каждой строке.
5. Дана таблица размера 5×10 заполненная целыми числами. В каждой строке найти среднее арифметическое всех элементов этой строки.
6. Дана таблица размера 5×10 заполненная целыми числами. Преобразовать эту таблицу, поменяв местами минимальный и максимальный элемент в каждой строке.
7. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное значение среди сумм элементов ее строк и столбцов и номер строки (столбца) с этим минимальным значением.
8. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное значение среди максимальных элементов каждой строки и столбца.
9. Дана таблица размера 6×10 заполненная целыми числами. Найти количество ее столбцов, все элементы которых различны.
10. Дана таблица размера 4×9 заполненная целыми числами. Вывести номера строк, содержащих одинаковые элементы.
11. Дана квадратная таблица порядка 5 заполненная целыми числами.
12. Найти суммы элементов расположенных на главной и побочной диагонали.
13. Дана квадратная таблица порядка 5 заполненная целыми числами. Заменить нулями элементы матрицы, лежащие ниже главной и выше побочной диагонали.
14. Дана квадратная таблица порядка 5 заполненная целыми числами. Заменить нулями элементы, лежащие одновременно выше главной диагонали (включая эту диагональ) и выше побочной диагонали (также включая эту диагональ).
15. Дана таблица размера 5×10 заполненная целыми числами. Вывести номера строк, элементы которых монотонно возрастают.
16. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальный среди элементов тех строк, которые упорядочены либо по возрастанию. Если такие строки отсутствуют, то вывести 0.
17. Даны два числа k_1 и k_2 и таблица размера 4×10 заполненная целыми числами. Поменять местами строки таблицы с номерами k_1 и k_2 .
18. Дана таблица размера 5×10 заполненная целыми числами. Поменять местами строки, содержащие минимальный и максимальный элементы таблицы.
19. Дана таблица размера 5×10 заполненная целыми числами. Поменять местами столбец с номером 1 и первый из столбцов, содержащих только положительные элементы.
20. Дано число k и таблица размера 4×10 заполненная целыми числами.
21. Удалить строку таблицы с номером k .

22. Дана таблица размера 5 x 10 заполненная целыми числами. Удалить столбец, содержащий минимальный элемент таблицы.
23. Дана таблица размера 5 x 10 заполненная целыми числами. Удалить первый столбец, содержащие только положительные элементы.
24. Дано число k и таблица размера 4 x 9 заполненная целыми числами.
25. Перед строкой таблицы с номером k вставить строку из нулей.
26. Дана таблица размера 4 x 9 заполненная целыми числами.
27. Продублировать строку таблицы, содержащий ее минимальный элемент.
28. Дана таблица размера 5 x 9 заполненная целыми числами. Перед столбцом, содержащим минимальный элемент, добавить столбец, состоящий из единиц.
29. Дана таблица размера 5 x 10 заполненная целыми числами. Вывести номера строк, элементы которых монотонно убывают.

Критерии оценивания заданий

Вопрос 1. ТЕОРЕТИЧЕСКОЕ ЗАДАНИЕ (max 20 баллов)

18-20 баллов ($k = 0,9-1,0$) ставится от максимального количества баллов, если обучающийся:

полно раскрыл содержание материала в объеме, предусмотренном рабочей программой, изложил материал грамотным языком в определенной логической последовательности, точно используя специальную терминологию;

правильно выполнил рисунки, чертежи, сопутствующие ответу; отвечал самостоятельно без наводящих вопросов преподавателя.

Возможны одна - две неточности при освещении второстепенных вопросов или в выкладках, которые студент легко исправил по замечанию преподавателя.

15-17 баллов ($k = 0,75-0,85$) ставится, если ответ удовлетворяет основным требованиям, но при этом имеет один из недостатков:

в изложении допущены небольшие пробелы, не искавшие содержание ответа;

допущены один – два недочета при освещении основного содержания ответа, исправленные по замечанию преподавателя;

допущена ошибка или имеется более двух недочетов при освещении второстепенных вопросов или в выкладках, легко исправленные по замечанию преподавателя.

10-14 баллов ($k = 0,5-0,7$) ставится в следующих случаях:

неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса;

имелись затруднения или допущены ошибки в определении понятий, использовании терминологии и выкладках (определениях), исправленные после нескольких наводящих вопросов преподавателя;

при знании теоретического материала выявлена недостаточная сформированность основных умений и навыков.

5-9 баллов ($k = 0,25-0,45$) ставится в следующих случаях: не раскрыто основное содержание учебного материала;

обнаружено незнание или непонимание обучающимся большей или наиболее важной части учебного материала;

допущены ошибки в определении понятий, при использовании терминологии, в выкладках, которые не исправлены после нескольких наводящих вопросов преподавателя.

меньше 5 баллов ($k = 0-0,25$) ставится, если:

обучающийся обнаружил полное незнание и непонимание изучаемого учебного материала или не смог ответить ни на один из поставленных вопросов по изучаемому материалу.

Вопрос 2 .ПРАКТИЧЕСКОЕ ЗАДАНИЕ (max 80 баллов: А – 40 баллов, В – 40 баллов)

36-40 баллов ($k = 0,9-1,0$) ставится если обучающийся: полностью выполнил все требования индивидуального задания;

отвечал самостоятельно без наводящих вопросов преподавателя. Возможны одна - две неточности при освещении второстепенных вопросов или в выкладках, которые студент легко исправил по замечанию преподавателя.

31-35 баллов ($k = 0,78 -0,88$) ставится, если ответ удовлетворяет основным требованиям, но при этом имеет один из недостатков:

в выполнении допущены небольшие неточности, не искажившие решение задания;

допущены один – два недочета при освещении основного содержания ответа, исправленные по замечанию преподавателя;

допущены ошибка или более двух недочетов при освещении второстепенных вопросов или в выкладках, легко исправленные по замечанию преподавателя.

26-30 баллов ($k = 0,65-0,75$) ставится в следующих случаях:

допущены неточности в выполнении индивидуального задания, но показано общее понимание вопроса;

имелись затруднения или допущены ошибки в выполнении индивидуального задания, но осуществлены значительные исправления после нескольких наводящих вопросов преподавателя;

20-25 баллов ($k = 0,5-0,63$) ставится в следующих случаях: не в полном объеме решена поставленная задача.