

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Чернецкая Ирина Евгеньевна
Должность: Заведующий кафедрой
Дата подписания: 20.09.2023 15:36:20
Уникальный программный ключ:
bdf214c64d8a381b0782ea566b0dce05e3f5ea2d

МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ

Заведующий кафедрой

вычислительной техники

 И.Е.Чернецкая

« 01 » сентября 2023 г.

ОЦЕНОЧНЫЕ СРЕДСТВА

для текущего контроля успеваемости

и промежуточной аттестации обучающихся

по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

(наименование дисциплины)

09.03.01 Информатика и вычислительная техника, направленность
(профиль) «Вычислительные машины, комплексы, системы и сети»

(код и наименование ОПОП ВО)

1 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ

1.1 ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ

Раздел (тема) дисциплины Общие сведения о языке Object Pascal и интегрированной среде Delphi (Lazarus)

1. Что представляют из себя линейные программы?
2. Для чего применяются ряды Тейлора?
3. Как производится расчет абсолютной и относительной погрешности при вычислении значения левой и правой части ряда Тейлора?
4. Что происходит с абсолютной и относительной погрешностями при увеличении числа слагаемых ряда?

Раздел (тема) дисциплины Базовые элементы и конструкции Object Pascal

1. Что представляют из себя разветвленные программы?
2. Что называется неполной альтернативой?
3. Что такое составной оператор и как он применяется совместно с операторами ветвлений?
4. Какие ключевые слова используются при записи условных ветвлений?

Раздел (тема) дисциплины Операторы Object Pascal

1. Какие операторы используются для описания линейных алгоритмов?
2. Какие операторы используются для описания разветвленных алгоритмов?
3. Какие операторы используются для описания циклических алгоритмов?
4. С помощью каких символов производится запись оператора присваивания?
5. Какие ключевые слова используются для описания разветвленных и циклических алгоритмов?

Раздел (тема) дисциплины Программирование циклических алгоритмов

1. Для чего применяются циклы в программах?
2. Какие типы циклов существуют? Как они записываются в конструкциях языков высокого уровня?
3. Что является условие завершения цикла каждого конкретного типа?
4. Какое минимальное и максимальное количество раз может быть выполнен цикл каждого из типов?
5. Что такое бесконечные циклы?
6. Какие средства языка применяются для досрочного прерывания тела цикла?
7. Какие средства языка применяются для досрочного перехода к следующей итерации цикла?

Раздел (тема) дисциплины Массивы

1. Для чего применяются массивы?
2. Какие типы массивов существуют?
3. Что такое размерность и размер массива? Чем отличаются указанные понятия?
4. В чем отличия массивов, хранимых в статической и динамической памяти?
5. Что такое операция индексации?
6. Что происходит при обращении к несуществующему элементу массива?

Раздел (тема) дисциплины Строки

1. Для чего применяются строки?
2. Как выбор кодировки влияет на двоичное содержимое строки?
3. Какие подпрограммы предназначены для обработки строк?

Раздел (тема) дисциплины Записи

1. Для чего применяются записи?
2. Как производится обращение к полям записей?
3. В чем заключается функционал оператора присоединения?
4. Как определяется размер записи в памяти компьютера? Как на него влияет выравнивание полей?
5. Что представляет собой вариантная часть записи?

Раздел (тема) дисциплины Процедуры и функции

1. Для чего применяются подпрограммы?
2. В чем отличие процедур от функций?
3. Для чего используются параметры?
4. Какие модификаторы применяются при передаче параметров?
5. Чем отличается передача параметров по ссылке и по значению?
6. Что такое параметры по умолчанию?
7. Чем отличается определение понятия рекурсия в математике и программировании?

Раздел (тема) дисциплины Множества

1. Что такое множество с точки зрения языка программирования?
2. Как производится хранение множества?
3. Какие операции применяются при работе с множествами?
4. Как реализовать операцию дополнения множества?
5. Какую максимальную мощность может иметь множество?

Раздел (тема) дисциплины Файлы

1. Как логически организовано хранение информации на жестком диске компьютера?
2. Какие типы файлов поддерживаются при работе с ними в Delphi?
3. Какие подпрограммы используются при работе с файлами?
4. В чем отличие работы с текстовыми и бинарными файлами?
5. Какие подпрограммы используются для сканирования файлов в заданном каталоге?

Раздел (тема) дисциплины Работа с графикой

1. Какие компоненты используются для вывода графической информации?
2. Дайте определение виртуального экрана?
3. Для чего используется свойство Canvas визуальных компонентов?
4. Для чего применяются кисть и карандаш?
5. С помощью каких подпрограмм можно задать цвет кисти или карандаша?

Раздел (тема) дисциплины Общие сведения о языках С и С++

1. В чем отличие языков С и С++?
2. К какому классу языков программирования относятся языки С и С++?
3. В чем ключевые отличия языков программирования С++ и Delphi?
4. В чем ключевые отличия языков программирования С++ и С#?
5. В каких языках программирования реализована поддержка множественного наследования?

Раздел (тема) дисциплины Типы данных С++

1. Какие категории типов данных имеются в языке С++?
2. Какие целые типы данных имеются в языке С++?
3. Какие вещественные типы данных имеются в языке С++?
4. Как производится объявление массивов в С++?
5. Как производится работа с динамическими массивами в С++?

Раздел (тема) дисциплины Определения и описания (объявления)

1. В чем отличие определения от описания?
2. Как производится описание переменных?
3. Как производится описание типов?
4. Как производится описание констант?
5. Какие типы файлов используются для хранения определений и описаний?

Раздел (тема) дисциплины Выражения

1. В чем отличие операторов и операций?
2. Какие операции используются при записи арифметических выражений?
3. В чем отличие поразрядных и логических операций?
4. Какой тип результата имеют арифметические операции при работе с целочисленными аргументами?
5. Скобки какого типа применяются для изменения порядка выполнения операций в арифметическом выражении?

Раздел (тема) дисциплины Операторы в C++

1. Какие операторы используются для описания линейных алгоритмов?
2. Какие операторы используются для описания разветвленных алгоритмов?
3. Какие операторы используются для описания циклических алгоритмов?
4. С помощью какого символа производится запись оператора присваивания?
5. Какие ключевые слова используются для описания разветвленных и циклических алгоритмов?

Раздел (тема) дисциплины Массивы

1. Для чего применяются массивы?
2. Какие типы массивов существуют?
3. Что такое размерность и размер массива? Чем отличаются указанные понятия?
4. В чем отличия массивов, хранимых в статической и динамической памяти?
5. Что такое операция индексации?
6. Что происходит при обращении к несуществующему элементу массива?

Раздел (тема) дисциплины Функции

1. Для чего применяются функции?
2. Как реализуется запись аналога процедуры?
3. Для чего используются параметры?
4. Какие модификаторы применяются при передаче параметров?
5. Чем отличается передача параметров по ссылке и по значению?

Раздел (тема) дисциплины Структуры и объединения

1. Дайте определение структуры.
2. Как производится запись объединений?
3. В чем отличие структуры от объединения?
4. Как производится применение ссылок при работе со структурами?
5. Что такое поле в составе структуры?

Раздел (тема) дисциплины Ввод-вывод в C++

1. Дайте определение файлового потока.
2. Какие файловые потоки используются для работы с файлами?
3. Какое пространство имен используется при работе с файловыми потоками?
4. Какие функции применяются для работы с файлами?

5. В чем отличие при работе с текстовыми и бинарными файлами?

Раздел (тема) дисциплины Ввод-вывод в С++

1. Что такое препроцессор?
2. На каком этапе производится работа препроцессора?
3. Какие ключевые слова управляют работой препроцессора?
4. Для чего применяется директива define?
5. Дайте определение шаблонов.

Шкала оценивания: 48-балльная.

Критерии оценивания

2 балла выставляется обучающемуся, если он принимает активное участие в беседе по большинству обсуждаемых вопросов (в том числе самых сложных); демонстрирует сформированную способность к диалогическому мышлению, проявляет уважение и интерес к иным мнениям; владеет глубокими (в том числе дополнительными) знаниями по существу обсуждаемых вопросов, ораторскими способностями и правилами ведения полемики; строит логичные, аргументированные, точные и лаконичные высказывания, сопровождаемые яркими примерами; легко и заинтересованно откликается на неожиданные ракурсы беседы; не нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

1 балл выставляется обучающемуся, если он принимает участие в обсуждении не менее 50% дискуссионных вопросов; проявляет уважение и интерес к иным мнениям, доказательно и корректно защищает свое мнение; владеет хорошими знаниями вопросов, в обсуждении которых принимает участие; умеет не столько вести полемику, сколько участвовать в ней; строит логичные, аргументированные высказывания, сопровождаемые подходящими примерами; не всегда откликается на неожиданные ракурсы беседы; не нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

0 баллов (выставляется обучающемуся, если он не владеет содержанием обсуждаемых вопросов или допускает грубые ошибки; пассивен в обмене мнениями или вообще не участвует в дискуссии; затрудняется в построении монологического высказывания и (или) допускает ошибочные высказывания; постоянно нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

2 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ

2.1 ТЕМЫ КУРСОВЫХ РАБОТ

1. В заданном ориентированном графе найти кратчайший путь между парой указанных вершин.
2. В заданном неориентированном графе найти все пути между указанной парой вершин. Повторное посещение уже пройденных вершин в путях не допускается.
3. Гамильтонов путь – это путь, однократно проходящий через все вершины графа. В заданном ориентированном графе найти кратчайший гамильтонов путь, соединяющий указанную пару вершин.
4. В заданном неориентированном графе найти все гамильтоновы пути, соединяющие указанную пару вершин.
5. Гамильтонов цикл – это цикл, однократно проходящий через все вершины графа. В заданном неориентированном графе найти кратчайший гамильтонов цикл.
6. В заданном ориентированном графе найти все гамильтоновы циклы.
7. Хроматическим числом неориентированного графа называется минимальное число цветов, в которое можно раскрасить вершины графа так, чтобы соединенные ребром вершины были раскрашены в разные цвета. Определить хроматическое число для заданного неориентированного графа.
8. Найти минимальную раскраску заданного неориентированного графа (см. предыдущее задание).
9. Графы называются изоморфными, если из одного можно получить другой путем перенумерации его вершин. Определить, является ли заданная пара графов изоморфной.
10. Определить, содержит ли заданный граф G подграф \tilde{G} , изоморфный графу G' .
11. Для графов G_1 и G_2 найти максимальный по включению изоморфный подграф \tilde{G} .
12. Автоморфизмом графа $G = \langle A, V \rangle$ называется такая перестановка σ его вершин, в ходе которой получается граф $G' = \langle \sigma(A), V \rangle$, изоморфный графу G . Найти все автоморфизмы заданного графа G .
13. Для неориентированного графа заданы веса ребер. Найти такое разбиение графа на N непустых подграфов, чтобы сумма весов ребер, связывающих полученные подграфы, была минимальна.
14. Максимальным независимым множеством называется такое максимальное по включению подмножество вершин графа, в котором ни одна пара вершин не соединена ребром. Для заданного графа найти максимальное независимое множество.
15. Для заданного неориентированного графа найти все полносвязные подграфы из N вершин.
16. Для заданного неориентированного графа найти максимальный по включению полносвязный подграф.
17. Для заданного дерева найти все поддеревья, входящие в его состав. Определить их число.
18. Для заданного графа найти все связные подграфы, входящие в его состав. Определить их число.
19. Диаметр графа – это кратчайшее расстояние между парой наиболее удаленных друг от друга вершин графа. Для заданного графа определить его диаметр.
20. Граф называется связным, если для каждой пары вершин существует соединяющий их путь. Определить, является ли заданный неориентированный граф связным.

Шкала оценивания курсовых работ (или курсовых проектов): 100-балльная.

Критерии оценивания (нижеследующие критерии оценки являются примерными и могут корректироваться):

85–100 баллов (или оценка «отлично») выставляется обучающемуся, если тема курсовой работы раскрыта полно и глубоко, при этом убедительно и аргументированно изложена собственная позиция автора по рассматриваемому вопросу; курсовая работа демонстрирует способность автора к сопоставлению, анализу и обобщению; структура курсовой работы четкая и логичная; изучено большое количество актуальных источников, включая дополнительные источники, корректно сделаны ссылки на источники; самостоятельно подобраны убедительные примеры; основные положения доказаны; сделан обоснованный и убедительный вывод; сформулированы мотивированные рекомендации; выполнены требования к оформлению курсовой работы.

70–84 баллов (или оценка «хорошо») выставляется обучающемуся, если тема курсовой работы раскрыта, сделана попытка самостоятельного осмысления темы; структура курсовой работы логична; изучены основные источники, правильно оформлены ссылки на источники; приведены уместные примеры; основные положения и вывод носят доказательный характер; сделаны рекомендации; имеются незначительные погрешности в содержании и (или) оформлении курсовой работы.

50–69 баллов (или оценка «удовлетворительно») выставляется обучающемуся, если тема курсовой работы раскрыта неполно и (или) в изложении темы имеются недочеты и ошибки; отмечаются отступления от рекомендованной структуры курсовой работы; количество изученных источников менее рекомендуемого, сделаны ссылки на источники; приведены самые общие примеры или недостаточное их количество; вывод сделан, но имеет признаки неполноты и неточности; рекомендации носят формальный характер; имеются недочеты в содержании и (или) оформлении курсовой работы.

0–49 баллов (или оценка «неудовлетворительно») выставляется обучающемуся, если тема курсовой работы не раскрыта и (или) в изложении темы имеются грубые ошибки; структура курсовой работы нечеткая или не определяется вообще; количество изученных источников значительно менее рекомендуемого, неправильно сделаны ссылки на источники или они отсутствуют; не приведены примеры или приведены неверные примеры; отсутствует вывод или автор испытывает затруднения с выводами; не соблюдаются требования к оформлению курсовой работы.

2.2 БАНК ВОПРОСОВ И ЗАДАНИЙ В ТЕСТОВОЙ ФОРМЕ

1 Вопросы в закрытой форме

1.1. Что будет выведено на экран в результате выполнения программы:

```
var i,j: integer; x,y: real;  
begin i:=5; j:=2; x:=3.1; y:=x*j; i:=i mod j; x:=i / j; j:=j + i div j; writeln(i,j,x,y) end.
```

Варианты ответа:

	i	j	x	y
a	5	2	3.1	6.2
b	1	2	0	6.2
c	1	2	0.5	6.2
d	2	2	0.5	6.2

1.2. Что будет выведено на экран в результате выполнения программы:

```
var v,w,p: byte;  
begin v:=3; w:=2; p:=21; v:=p shl w; p:=not p shl w; w:=v and p; writeln(v,w,p) end.
```

Варианты ответа:

	v	w	p
a	42	4	86
b	84	4	172
c	84	8	86
d	84	8	172

1.3. Что будет выведено на экран в результате выполнения программы:

```
var x,y,z,w: boolean;
begin x:=true; y:=true; z:=false; z:=not x and y or z; w:=y or z xor y; y:=x and not y and z;
x:=y xor z; writeln(x,y,z,w) end.
```

Варианты ответа:

	x	y	z	w
a	false	false	false	false
b	false	true	false	false
c	true	true	false	false
d	true	false	true	false

1.4. Каким будет результат записи выражения по правилам Паскаля:

```
a: y:=(1+8*sqr(cos(x))*ln(tan(x))/2*arcsin(x));
b: y:=(1+8sqr(cos(x))*log2(sin(x)/cos(x))/(2*arctan(x/sqrt(1-sqr(x))));
c: y:=(1+8*sqr(cos(x))*ln(abs(sin(x)/cos(x)))/ln(2))/2/arctan(x/sqrt(1-sqr(x)));
d: y:=(1+8*sqr(cos(x))*ln(abs(sin(x)/cos(x)))/ln(2)/(2*arctan(sqrt(1-sqr(x))/x));
```

1.5. Что будет выведено на экран в результате выполнения программы:

```
var x,y: real; i,j: integer;
begin
  x:=sqrt(abs(-4))+0.1;
  i:=succ(round(x));
  y:=int(x)+frac(x);
  j:=ord(add(i))+i;
  writeln(x,y,i,j)
end.
```

Варианты ответа:

	x	y	i	j
a	2.1	2.1	3	4
b	16.1	16.1	17	18
c	2.1	2.1	2	3
d	2.1	2.1	3	3

1.6. Что будет выведено на экран в результате выполнения программы:

```
var x,y,z: integer;
begin
  x:=1; y:=2; z:=3;
  if x<y then x:=y else y:=x; if x=y then x:=z else y:=z;
  writeln(x,y,z)
end.
```

Варианты ответа:

	x	y	z
a	2	2	3

b	3	2	3
c	3	3	3
d	2	3	3

1.7. Что будет выведено на экран в результате выполнения программы:

```
var v,w: integer; x,y,z: real;
begin v:=12; w:=4; x:=0.5; y:=2.5e-1; v:=w div v; w:=w mod 3; z:=x/y; x:=x*z; y:=x+z-y;
writeln(v,w,x,y,z) end.
```

Варианты ответа:

	v	w	x	y	z
a	0	0	1	2.75	2
b	3	0	1	2.75	2
c	3	1	0.5	2.75	2
d	0	1	1	2.75	2

1.8. Что будет выведено на экран в результате выполнения программы:

```
var i,j,k: byte;
begin i:=2; j:=4; k:=j shl i; i:=j and k and i; j:=not i xor j; writeln(i,j,k) end.
```

Варианты ответа:

	i	j	k
a	255	251	32
b	255	251	16
c	0	251	16
d	0	4	16

1.9. Что будет выведено на экран в результате выполнения программы:

```
var i,j,k: boolean;
begin i:=false; j:=true; k:=not i xor j; i:=i or j and k; j:=j xor not i and k; writeln(i,j,k) end.
```

Варианты ответа:

	i	j	k
a	false	true	true
b	true	true	false
c	false	false	true
d	false	true	false

1.10. Что будет выведено на экран в результате выполнения программы:

```
var m,n,k: integer;
begin m:=0; n:=2; k:=4; if m>n then if n<k then n:=k else k:=m else m:=k; writeln(m,n,k) end.
```

Варианты ответа:

	m	n	k
a	4	2	4
b	0	2	0
c	0	4	4
d	0	2	4

1.11. Что будет выведено на экран в результате выполнения программы:

```
var i,n,k: integer;
begin
```

```

k:=1; n:=1;
for i:=1 to 5 do begin
if odd(k) then n:=n+1 else n:=n+2;
k:=k+i
end;
writeln(i,n,k)
end.

```

Варианты ответа:

	i	n	k
a	6	8	16
b	6	7	16
c	6	8	15
d	6	8	17

1.12. Что будет выведено на экран в результате выполнения программы:

```

var i,n,k: integer;
begin
n:=10; i:=0; k:=1;
repeat
i:=i+k; k:=k+1;
if not odd(k) then n:=n+1;
until i>n;
writeln(i,n,k)
end.

```

Варианты ответа:

	i	n	k
a	14	13	5
b	14	13	6
c	15	13	6
d	15	12	6

1.13. Дан текст программы на Паскале:

```

var i,n,k: integer;
begin
n:=1; i:=5; k:=10;
while true do begin
n:=n*i; k:=k-1;
if n>i+k then break;
i:=i+1;
end;
writeln(i,n,k);
end.

```

Какое из перечисленных утверждений будет верно после ее выполнения:

- a: $n < i$ и $n < k$
- b: $i < n$ и $i < k$
- c: $i < n$ и $n < k$
- d: $n < i$ и $i < k$

1.14. Что будет выведено на экран в результате выполнения программы:

```
var a: array[1..10] of byte = (1,3,5,7,9,11,13,15,17,19);
var i,j,k: byte;
begin
i:=low(a)+2;
while i<=high(a) do begin
j:=i-1; k:=j-1;
a[i]:=a[j]+a[k];
i:=i+1;
end;
writeln(a[high(a)]);
end.
```

Варианты ответа:

- a: 19
- b: 100
- c: 122
- d: 123

1.15. Что будет выведено на экран в результате выполнения программы:

```
const Digits: array[0..10] of Char = '0123456789'#0;
var i: byte = 5; j: byte = 3;
begin
writeln( Digits + i - j );
end.
```

Варианты ответа:

- a: 23456789
- b: '23456789'
- c: 3456789
- d: 123456789

1.16. Что будет выведено на экран в результате выполнения программы:

```
var Digits: array[0..10] of Char = '0123456789'#0;
var i,j: byte;
begin
i:=StrLen(Digits)-1;
repeat dec(i); Digits[i-1]:=Digits[i] until i=1;
writeln( Digits );
end.
```

Варианты ответа:

- a: 9999999999
- b: 9999999998
- c: 8888888889
- d: 8888888888

1.17. Что будет выведено на экран в результате выполнения программы:

```
type
TMonth = (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec);
TMonths = set of TMonth;
const summer: TMonths = [Jun..Aug];
var ms: TMonths = []; i: TMonth;
```

```

begin
Include(ms,Jul); Include(ms,Sep);
if ms <= summer then i:=pred(Jun) else
if summer <= ms then i:=succ(Mar) else i:=pred(Dec);
writeln(ord(i))
end.

```

Варианты ответа:

- a: 11
- b: 10
- c: 9
- d: Nov

1.18. Что будет выведено на экран в результате выполнения программы:

```

procedure P(const a: real; var b: real; out c: real);
begin b:=2*a*b; c:=0.5*a*b end;
var x,y,z: real;
begin
    x:=3.5; y:=-1.5; z:=1.33;
    P(x,y,z);
    writeln(x,y,z)
end.

```

Варианты ответа:

- | | x | y | z |
|---|------|-------|-------|
| a | 3.5 | -10.5 | 1.33 |
| b | 3.5 | -10.5 | 0.0 |
| c | 3.5 | -10.5 | -18.4 |
| d | -3.5 | -10.5 | -18.4 |

1.19. Какая пара значений будет выведена на экран в результате выполнения программы:

```

function F(var n: integer): integer;
begin
    if n<=0 then begin n:=1; result:=0; exit end;
    if odd(n) then begin inc(n); result:=n*n; exit end;
    result:=2*n
end;
var n: integer = 5;
begin
    writeln(F(n)+F(n),n)
end.

```

Варианты ответа:

- a: 46 6
- b: 48 5
- c: 46 5
- d: 48 6

1.20. Что будет выведено на экран в результате выполнения программы:

```

function Q(const x: integer): real;
begin
    if x <= 0 then begin result:=1; exit end;

```

```
    if x in [1,2] then begin result:=Q(x-1)*0.5; exit end;  
    result:=2.0*Q(x-1);  
end;  
begin writeln(Q(5)) end.
```

Варианты ответа:

- a: 0.25
- b: 0.5
- c: 1.0
- d: 2.0

1.21. Допускается ли задание начальных значений глобальным переменным в разделе описаний программы?

Варианты ответа:

- a: да
- b: нет
- c: зависит от установки опций компилятора

1.22. Для преждевременного прерывания цикла используется

Варианты ответа:

- a: break
- b: continue
- c: jump

1.23. Для преждевременного перехода к следующей итерации цикла используется

Варианты ответа:

- a: continue
- b: break
- c: jump
- d: вложение цикла в еще один цикл

1.24. В каких случаях оправдано использование оператора goto?

Варианты ответа:

- a: для выхода из многократно вложенных циклов
- b: для организации циклов
- c: для организации условных ветвлений
- d: при организации перегрузки функций

1.25. Массивы какой минимальной размерности поддерживаются в Delphi?

Варианты ответа:

- a: 1
- b: 0
- c: 2
- d: не ограничено

1.26. Массивы какой максимальной размерности поддерживаются в Delphi?

Варианты ответа:

- a: 1
- b: 0
- c: 2
- d: не ограничено

1.27. Допускается ли задание начальных значений глобальным переменным в разделе описаний программы?

Варианты ответа:

- a: не ограничено
- b: 1
- c: 0
- d: 2

1.28. Размер динамического массива

Варианты ответа:

- a: определяется в процессе выполнения программы
- b: известен на этапе компиляции
- c: динамически меняется компилятором прозрачно для пользователя

1.29. Размер статического массива

Варианты ответа:

- a: известен на этапе компиляции
- b: определяется в процессе выполнения программы
- c: динамически меняется компилятором прозрачно для пользователя

1.30. Может ли динамический массив быть многомерным?

Варианты ответа:

- a: да
- b: нет
- c: может, только если память выделяется с помощью GetMem() или SetLength()

2 Вопросы в открытой форме

- 2.1. Процессор исполняет программы в виде ...
- 2.2. Компилятор предназначен для ...
- 2.3. Линкер (компоновщик) предназначен для ...
- 2.4. К исполняемым файлам относятся файлы ...
- 2.5. К файлам с исходными кодами программы на Delphi относятся файлы ...
- 2.6. Оптимизация программы – это ...
- 2.7. Интегрированная среда разработки (IDE) Delphi включает ...
- 2.8. VCL – это ...
- 2.9. VCL содержит компоненты для ...
- 2.10. Окно инспектора объектов используется для ...
- 2.11. Самый близкий аналог свойства компонента – это ...
- 2.12. Самый близкий аналог события компонента – это ...
- 2.13. Delphi может быть использована для создания ...
- 2.14. Заголовок проекта исполняемого файла .exe начинается служебным словом ...
- 2.15. Для организации ввода/вывода в консольных приложениях используется пара

процедур ...

- 2.16. Какой объект присутствует в любом оконном VCL-приложении?
- 2.17. Возможно ли использование инспектора объектов для редактирования свойств элементов консольного приложения?
- 2.18. Для вывода текстовой информации на форму обычно используют компоненты ...
- 2.19. Для ввода информации в оконных приложениях обычно используют компоненты ...
- 2.20. Для обработки события нажатия мышки в оконном VCL приложении используется обработчик ...
- 2.21. К зарезервированным словам относятся ...
- 2.22. Некорректным идентификатором в Delphi является ...
- 2.23. К операторам относится ...
- 2.24. Список подключаемых модулей начинается ключевым словом ...
- 2.25. Переменные описываются с помощью ключевого слова ...
- 2.26. Типы описываются с помощью ключевого слова ...
- 2.27. Константы описываются с помощью ключевого слова ...
- 2.28. К целым типам относится ...
- 2.29. Какие типы присутствуют среди predefined типов Delphi и не содержат знака?
- 2.30. Для большинства программ оптимальным типом с плавающей точкой является ...

3 Установление правильной последовательности

3.1. В каком порядке производится разработка программного обеспечения?

- анализ требований
- составление технического задания
- проектирование структур данных
- разработка алгоритмов
- написание кода
- отладка
- тестирование
- сопровождение

3.2. В каком порядке описываются элементы заголовка процедуры?

- ключевое слово procedure
- имя процедуры
- параметры
- директивы
- локальные объявления
- тело процедуры

3.3. В каком порядке описываются элементы заголовка функции?

- ключевое слово function
- имя процедуры
- параметры
- директивы
- локальные объявления
- тело процедуры

3.4. В каком порядке указываются ключевые слова, обозначающие элементы модуля (unit)?

- unit
- interface
- implementation

- initialization
- finalization

3.5. Расположите типы памяти в порядке увеличения времени доступа к ним.

- регистры процессора
- кэш-память первого уровня (L1)
- кэш-память второго уровня (L2)
- кэш-память третьего уровня (L3), при наличии
- оперативная память
- жесткий диск

3.6. Расположите в порядке запуска элементы компилятора.

- препроцессор (при наличии)
- компилятор
- линкер (компоновщик)

3.7. Расположите обработчики событий формы оконного приложения в порядке их срабатывания.

- OnCreate()
- OnShow()
- OnClick()
- OnClose()

3.8. Расположите ключевые слова в том порядке, в котором они используются при записи консольной программы без подпрограмм.

- program
- uses
- var
- begin
- end

3.9. Расположите в нужном порядке лексемы языка при объявлении переменной.

- var
- имя переменной
- :
- тип переменной
- ;

3.10. Расположите типы данных в порядке увеличения их размера в байтах.

- Byte
- Word
- LongWord
- Int64

3.11. Расположите типы данных в порядке увеличения их размера в байтах.

- Single
- Double
- Extended

3.12. Расположите в нужном порядке лексемы языка при объявлении типа.

- type
- имя типа

- =
- описание типа
- ;

3.13. Расположите в нужном порядке лексемы языка при объявлении нетипизированной константы.

- const
- имя константы
- =
- значение константы
- ;

3.14. Расположите в нужном порядке лексемы языка при объявлении нетипизированной константы.

- const
- имя константы
- :
- тип константы
- =
- значение константы
- ;

3.15. Расположите в нужном порядке лексемы языка при объявлении переменной скалярного типа, имеющей начальное значение.

- var
- имя переменной
- :
- тип переменной
- =
- значение
- ;

3.16. Укажите порядок следования элементов при записи вещественной константы.

- знак мантиссы
- мантисса
- E
- знак порядка
- порядок

3.17. Укажите порядок записи лексем в операторе присваивания.

- леводопустимое выражение
- :=
- праводопустимое выражение
- ;

3.18. Укажите порядок выполнения арифметических действий в выражении $1*2 + \text{Round}(3.1 - x)$

- -
- Round
- *
- +

3.19. Расположите целочисленные операции в порядке увеличения длительности их выполнения процессором в тактах

- сложение
- умножение
- деление
- тригонометрия

3.20. Укажите порядок следования лексем при записи оператора условного ветвления.

- if
- условие
- then
- оператор 1
- else
- оператор 2
- ;

3.21. Укажите порядок следования лексем при записи оператора выбора.

- case
- выражение
- of
- значение выражения
- ;
- оператор
- end
- ;

3.22. Укажите порядок следования лексем при записи цикла с предусловием.

- while
- условие
- do
- оператор
- ;

3.23. Укажите порядок следования лексем при записи цикла с постусловием.

- repeat
- операторы
- ;
- until
- условие

3.24. Укажите порядок следования лексем при записи цикла со счетчиком.

- for
- переменная счетчик
- :=
- начальное значение счетчика
- to/downto
- конечное значение счетчика
- do
- оператор
- ;

3.25. Укажите порядок следования лексем при объявлении одномерного статического

массива.

- имя переменной
- :
- array
- [
- минимальное значение индекса
- ..
- максимальное значение индекса
-]
- of
- тип элементов
- ;

3.26. Укажите порядок следования лексем при объявлении одномерного динамического массива, поддерживаемого компилятором.

- имя переменной
- :
- array
- of
- тип элементов
- ;

3.27. Укажите порядок следования лексем при объявлении двумерного статического массива.

- имя переменной
- :
- array
- [
- диапазон изменения индекса 1
- ,
- диапазон изменения индекса 2
-]
- of
- тип элементов
- ;

3.28. Укажите порядок следования лексем при объявлении типа, представляющего собой множество десятичных цифр.

- type
- имя типа
- =
- set
- of
- 0
- ..
- 9
- ;

3.29. Укажите порядок следования лексем при объявлении переменной типа запись.

- имя переменной
- :
- record

- список полей
- :
- тип
- ;
- end;

3.30. Укажите порядок следования лексем при использовании оператора присоединения.

- with
- переменная типа запись
- do
- оператор
- ;

4 Установление соответствия

4.1. Укажите синтаксически неправильное значение константы

- +8.903E-17.1
- -123.17
- \$F0
- 1547

4.2. При делении целых чисел при помощи операции div результат получается

- целым
- вещественным
- символьным

4.3. При делении целых чисел при помощи операции / результат получается

- вещественным
- целым
- символьным

4.4. Операция mod обозначает

- остаток от деления
- модуль числа
- изменение (модифицирование) знака

4.5. Если $A = 1001\ 1100$ и $B = 1100\ 1111$, то $A \text{ and } B =$

- 1000 1100
- 1001 1100
- 1101 1111
- нет правильного ответа

4.6. Если $A = 1001\ 1100$ и $B = 1100\ 1111$, то $A \text{ or } B =$

- 1101 1111
- 1000 1100
- 1111 1101
- нет правильного ответа

4.7. Если $A = 1001\ 1100$ и $B = 1100\ 1111$, то $A \text{ xor } B =$

- 0101 0011
- 1101 1111
- 0110 1011
- нет правильного ответа

4.8. Если $A = 1001\ 1100$, то $\text{not } A =$

- 0110 0011
- 0011 1000
- 0110 1011
- нет правильного ответа

4.9. Если $A = 1001\ 1100$, то $A \text{ shl } 1 =$

- 0011 1000
 - 0001 1100
 - 0000 1110
 - нет правильного ответа
- 4.10. Если $A = 1001\ 1100$, то $A \text{ shr } 2 =$
- 0010 0111
 - 0100 1110
 - 0011 1000
 - нет правильного ответа
- 4.11. Какой символ НЕ используется в синтаксических конструкциях Delphi?
- !
 - @
 - \$
 - ^
- 4.12. Какое из действий обладает наивысшим приоритетом?
- вызов функции
 - умножение
 - сложение
 - сдвиги
- 4.13. Без чего невозможно произвести вызов функции?
- без указания имени функции
 - без получения значения результата
 - без указания соглашения о вызове
 - без передачи параметров
- 4.14. Возможна ли запись оператора if без ветви else?
- да
 - нет
 - зависит от установки опций компилятора
- 4.15. Для получения случайного числа используется функция
- random
 - rand
 - randomize
- 4.16. Укажите синтаксически неправильное значение константы
- 0xF0
 - -123.17
 - +8.903E-17
 - 1457
- 4.17. Укажите синтаксически неправильное значение константы
- -123,17
 - \$F0
 - +8.903E-17
 - 1457
- 4.18. Какая из перечисленных обратных тригонометрических функций поддерживается в Delphi по умолчанию (без подключения дополнительных модулей)?
- arctan
 - arcsin
 - arccos
- 4.19. Для преобразования числа с плавающей точкой в строку обычно не используют
- IntToStr
 - FloatToStr
 - FloatToStrF
 - Format

4.20. Операторы отделяются друг от друга символом

- ;
- .
- :
- ...

4.21. Для организации ветвлений не подходит оператор

- else
- if
- case
- goto

4.22. Возможно ли вложение операторов друг в друга

- да
- нет
- зависит от установки опций компилятора

4.23. Для организации альтернативных ветвлений с более чем 2 ветвями не используют

- return
- совокупность операторов if
- case

4.24. Резервированное слово else НЕ может встречаться

- отдельно от других операторов
- совместно с if
- совместно с case

4.25. Для организации цикла с заранее известным числом итераций лучше всего подходит оператор

- for
- while
- repeat
- loop

4.26. Какой из циклов всегда выполняется хотя бы 1 раз?

- repeat
- for
- while
- loop

4.27. Какой из циклов допускает отсутствие операторных скобок?

- repeat
- for
- while
- loop

4.28. Какая конструкция языка используется при необходимости записи нескольких операторов, когда правилами языка разрешен только один?

- begin/end
- while/do/end
- case/of/end
- if/then/else

4.29. Какой из циклов является бесконечным?

- repeat ... until False;
- while False do ...;
- for I:=1 to N do ... Dec(I) ...;
- любой цикл с использованием меток (label)

4.30. При организации цикла for со счетчиком -1 необходимо использовать ключевое слово

- downto

- to
- dec
- inc

Шкала оценивания результатов тестирования: в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по заочной форме обучения – 60 баллов (установлено положением П 02.016).

Максимальный балл за тестирование представляет собой разность двух чисел: максимального балла по промежуточной аттестации для данной формы обучения (36 или 60) и максимального балла за решение компетентностно-ориентированной задачи (6). Балл, полученный обучающимся за тестирование, суммируется с баллом, выставленным ему за решение компетентностно-ориентированной задачи. Общий балл по промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по 5-балльной шкале следующим образом.

Соответствие 100-балльной и 5-балльной шкал

<i>Сумма баллов по 100-балльной шкале</i>	<i>Оценка по 5-балльной шкале</i>
100-85	отлично
84-70	хорошо
69-50	удовлетворительно
49 и менее	неудовлетворительно

2.3 КОМПЕТЕНТНОСТНО-ОРИЕНТИРОВАННЫЕ ЗАДАЧИ

1. Вывести на экран все простые числа в диапазоне от 1 до N . Число называется простым, если оно делится без остатка только само на себя и на 1.
2. Найти натуральное число в диапазоне от 1 до N с максимальной суммой делителей.
3. Представить заданное число в виде произведения простых чисел (например, $5040 = 2^4 \cdot 3^2 \cdot 5 \cdot 7$).
4. Найти наименьшее натуральное число N из диапазона $[A, B]$, представимое в виде суммы кубов двух натуральных чисел $N = x^3 + y^3$.
5. Натуральное число из N цифр называется числом Армстронга, если оно может быть представлено в виде суммы его цифр, возведенных в N -ую степень (например, $153 = 1^3 + 5^3 + 3^3$). Найти числа Армстронга в заданном диапазоне.
6. Указать все способы представления заданного числа N в виде суммы квадратов 3 натуральных чисел $N = x^2 + y^2 + z^2$.
7. Согласно теореме Лагранжа о сумме четырех квадратов любое число можно представить в виде суммы не более чем 4 квадратов натуральных чисел. Проверить справедливость теоремы для чисел из заданного диапазона.
8. (Плоские числа) Найти и вывести на экран все числа, меньшие N , единственным образом представимые в виде произведения простых чисел A и B (например, $985 = 5 \cdot 197$).
9. (Телесные числа) Найти и вывести на экран все числа, меньшие N , единственным образом представимые в виде произведения простых чисел A , B и C (например, $385 = 5 \cdot 7 \cdot 11$).
10. В приведенной последовательности сумм

$$S_1 = 1,$$

$$S_2 = 3 + 5 = 8,$$

$$S_3 = 7 + 9 + 11 = 27,$$

...

вычислить значение i -ой суммы. Вывести на экран все промежуточные значения.

11. Найти наилучшее приближение к заданному иррациональному числу (π , e , $\sqrt{2}$ и т.д.) в виде рациональной дроби $\frac{A}{B}$ в заданном диапазоне ($1 \leq A \leq N$, $1 \leq B \leq N$).

12. Проверить справедливость соотношения $\sum_{i=0}^n C_n^i = 2^n$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

13. Проверить справедливость соотношения $\sum_{i=0}^n (-1)^i C_n^i = 0$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

14. (Тождество Абеля) Убедиться в справедливости соотношения $2(n-1)n^{n-2} = \sum_{\substack{k+m=n, \\ k>1, m>1}} C_n^k k^{k-1} m^{m-1}$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

15. Проверить справедливость соотношения $n! = 1 + \sum_{i=1}^{n-1} i \cdot i!$.

16. Проверить справедливость соотношения $\sum_{i=1}^n i^3 = \left(\sum_{i=1}^n i \right)^2 = \frac{n^2(n+1)^2}{4}$.

17. Проверить справедливость соотношения $\sum_{k=0}^{n-1} 2^k = 2^n - 1$.

18. Проверить справедливость соотношения $(x - a)^n \equiv x^n - a \pmod{n}$, где x, a, n – целые положительные числа, n – простое число, $a \leq x$, с использованием датчика случайных чисел.

19. (малая теорема Ферма) Если a не делится на простое число p , то $a^{p-1} - 1$ делится на p . Проверить справедливость теоремы с использованием датчика случайных чисел.

20. Проверить или опровергнуть гипотезу Ферма. Простые числа вида $4k+1$ (k – натуральное число) можно единственным способом представить в виде суммы квадратов двух целых чисел. Например, при $k=1$ $p=4 \cdot 1 + 1 = 5 = 1^2 + 2^2$, а при $k=3$ $p=4 \cdot 3 + 1 = 13 = 2^2 + 3^2$.

21. Найти все простые множители заданного числа N .

22. Для всех нечетных чисел n значение $n^2 - 1$ делится на 8. Проверить справедливость утверждения.

23. Если $p > 3$ – простое число, то значение $p^2 - 1$ делится на 24. Убедиться в справедливости утверждения.

24. Найти все возможные значения цифр $0 \leq D, E, M, N, O, R, S, Y \leq 9$ такие, что выполняется равенство

$$\begin{array}{r} SEND \\ + MORE \\ \hline MONEY \end{array}$$

Пример решения: $D = 0, E = 0, M = 1, N = 0, O = 0, R = 0, S = 9, Y = 0$.

25. Убедиться в справедливости равенства $\sum_{i=1}^n H_i = (n+1)[H_{n+1} - 1]$, где $H_n = \sum_{k=1}^n \frac{1}{k}$ – ряд гармонических чисел.

26. Убедиться в справедливости равенства $\sum_{i=1}^n iH_i = C_{n+1}^2 \left(H_{n+1} - \frac{1}{2} \right)$, где $C_n^m = \frac{n!}{m!(n-m)!}$

– число сочетаний из n по m , $H_n = \sum_{k=1}^n \frac{1}{k}$ – сумма ряда гармонических чисел.

27. Убедиться в справедливости равенства $\sum_{i=1}^n C_i^k H_i = C_{n+1}^{k+1} \left(H_{n+1} - \frac{1}{k+1} \right)$ для заданных

значений n и k , где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m , $H_n = \sum_{k=1}^n \frac{1}{k}$ – сумма ряда

гармонических чисел.

28. Проверить справедливость соотношения $\sum_{k=0}^n x^k = \frac{x^{n+1} - 1}{x - 1}$.

29. Проверить справедливость соотношения $\sum_{k=1}^n \frac{1}{k(k+1)} = \frac{n}{n+1}$.

30. Проверить справедливость соотношения $\sum_{k=1}^n k \cdot 2^k = (n-1) \cdot 2^{n+1} + 2$.

Шкала оценивания решения компетентностно-ориентированной задачи: в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по очно-заочной и заочной формам обучения – 60 (установлено положением П 02.016).

Максимальное количество баллов за решение компетентностно-ориентированной задачи – 6 баллов.

Балл, полученный обучающимся за решение компетентностно-ориентированной задачи, суммируется с баллом, выставленным ему по результатам тестирования.

Общий балл промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по 5-балльной шкале.

Критерии оценивания решения компетентностно-ориентированной задачи:

6-5 баллов выставляется обучающемуся, если решение задачи демонстрирует глубокое понимание обучающимся предложенной проблемы и разностороннее ее рассмотрение; свободно конструируемая работа представляет собой логичное, ясное и при этом краткое, точное описание хода решения задачи и формулировку доказанного, правильного вывода (ответа); задача решена в установленное преподавателем время или с опережением времени.

4-3 балла выставляется обучающемуся, если решение задачи демонстрирует понимание обучающимся предложенной проблемы; задача решена типовым способом в установленное преподавателем время; имеют место несущественные недочеты в описании хода решения и (или) вывода (ответа).

2-1 балла выставляется обучающемуся, если решение задачи демонстрирует поверхностное понимание обучающимся предложенной проблемы; осуществлена попытка шаблонного решения задачи, но при ее решении допущены ошибки и (или) превышено установленное преподавателем время.

0 баллов выставляется обучающемуся, если решение задачи демонстрирует непонимание обучающимся предложенной проблемы и (или) задача не решена.

2.4 БАНК ВОПРОСОВ ДЛЯ КОНТРОЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ И ПРАКТИЧЕСКИХ РАБОТ

Лабораторные работы

Программирование линейных алгоритмов

1. Что представляют из себя линейные программы?
2. Для чего применяются ряды Тейлора?
3. Как производится расчет абсолютной и относительной погрешности при вычислении значения левой и правой части ряда Тейлора?
4. Что происходит с абсолютной и относительной погрешностями при увеличении числа слагаемых ряда?

Программирование разветвленных алгоритмов

1. Что представляют из себя разветвленные программы?
2. Что называется неполной альтернативой?
3. Что такое составной оператор и как он применяется совместно с операторами ветвлений?
4. Какие ключевые слова используются при записи условных ветвлений?

Работа с массивами

1. Для чего применяются массивы?
2. Какие типы массивов существуют?

3. Что такое размерность и размер массива? Чем отличаются указанные понятия?
4. В чем отличия массивов, хранимых в статической и динамической памяти?
5. Что такое операция индексации?
6. Что происходит при обращении к несуществующему элементу массива?

Записи и файлы

1. Для чего применяются записи?
2. Как производится обращение к полям записей?
3. В чем заключается функционал оператора присоединения?
4. Как определяется размер записи в памяти компьютера? Как на него влияет выравнивание полей?
5. Как логически организовано хранение информации на жестком диске компьютера?
6. Какие типы файлов поддерживаются при работе с ними в Delphi?
7. Какие подпрограммы используются при работе с файлами?

Программирование циклических алгоритмов

1. Для чего применяются циклы в программах?
2. Какие типы циклов существуют? Как они записываются в конструкциях языков высокого уровня?
3. Что является условием завершения цикла каждого конкретного типа?
4. Какое минимальное и максимальное количество раз может быть выполнен цикл каждого из типов?
5. Что такое бесконечные циклы?
6. Какие средства языка применяются для досрочного прерывания тела цикла?
7. Какие средства языка применяются для досрочного перехода к следующей итерации цикла?

Программирование подпрограмм. Рекурсия

1. Для чего применяются подпрограммы?
2. В чем отличие процедур от функций?
3. Для чего используются параметры?
4. Какие модификаторы применяются при передаче параметров?
5. Чем отличается передача параметров по ссылке и по значению?
6. Что такое параметры по умолчанию?
7. Чем отличается определение понятия рекурсия в математике и программировании?

Обработка строк

1. Для чего применяются строки?
2. Как выбор кодировки влияет на двоичное содержимое строки?

3. Какие подпрограммы предназначены для обработки строк?

Практические работы

Программирование на языках высокого уровня

1. Какие типы проектов существуют в средах программирования, используемых для создания приложений?
2. Как создать многомодульную программу?
3. Что такое компилятор?
4. Что такое линкер?
5. Что такое препроцессор?