

Санкт-Петербургское государственное бюджетное профессиональное
образовательное учреждение
«Академия управления городской средой, градостроительства и печати»



УТВЕРЖДАЮ

Заместитель директора по УМР

О.В.Фомичева

12 2023 г

Контрольно-оценочные средства
для текущего контроля и промежуточной аттестации
«ОП.06 ОСНОВЫ АЛГОРИТМИЗАЦИИ И
ПРОГРАММИРОВАНИЯ»

для специальности

08.02.15 Информационное моделирование в строительстве

Форма обучение – очная

Санкт-Петербург

2023 г.

Разработаны на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования 08.02.15 Информационное моделирование в строительстве, в соответствии с рабочей программой учебной дисциплины ОП.06 Основы алгоритмизации и программирования

Разработчики:

Разработчики: Ипатова С.В., Оболенская Е.Г. методисты СПб ГБПОУ «АУГСГиП»

Одобрено на заседании цикловой комиссии
Математики и информационных технологий

Протокол № 2

« 24 » 11 _____ 20 23 г.

Председатель цикловой комиссии

И.А. Минько /Минько И.А./

КОС соответствует ФГОС СПО

Эксперт

Минько И.А., преподаватель СПб ГБПОУ «Академия управления городской средой, градостроительства и печати»

Дата _____

Подпись _____

1. Результаты освоения учебной дисциплины, подлежащие проверке

Код ПК, ОК	Умения	Знания
ОК 01 -02 ОК 09 ПК 1.1; ПК 1.5 ПК 2.2- 2.3 ПК 3.1- 3.3 ЛР1-4, ЛР10, ЛР13-17	– работать в среде программирования; – использовать языки программирования	– типы данных; – базовые конструкции изучаемых языков программирования; – интегрированные среды программирования на изучаемых языках.

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

ОК 09 Пользоваться профессиональной документацией на государственном и иностранном языках

ПК 1.1. Адаптировать программные средства в соответствии со стандартами применения технологий информационного моделирования зданий

ПК 1.5. Автоматизировать решение задач формирования, анализа и передачи данных о здании средствами программ информационного моделирования

ПК 2.2 Проектировать строительные конструкции с использованием технологии информационного моделирования

ПК 2.3 Проектировать инженерные сети и оборудование с использованием технологии информационного моделирования

ПК 3.1. Формировать данные структурных элементов информационной модели при решении профильных задач на этапе разработки архитектурной, конструктивной частей, инженерных систем и оборудования проекта

ПК 3.2. Обработать данные структурных элементов информационной модели при решении профильных задач на этапе разработки архитектурной, конструктивной частей, инженерных систем и оборудования проекта

ПК 3.3. Актуализировать данные структурных элементов информационной модели при решении профильных задач на этапе разработки архитектурной, конструктивной частей, инженерных систем и оборудования проекта

2. Формы текущего контроля и оценивания по учебной дисциплине

Раздел/ тема	Форма текущего контроля	Оцениваемые У/З/ПК, ОК
Раздел 1. Основные принципы алгоритмизации и программирования	Тестовый контроль	ОК 01 -02 ОК 09 ПК 1.1 ПК 1.5 ПК 2.2- 2.3 ПК 3.1- 3.3 ЛР1-4, ЛР10, ЛР13-17
Раздел 2. Язык программирования	Тестовый контроль	
Раздел 3. Основы объектно-ориентированного программирования	Тестовый контроль контрольная практическая контрольная работа	
Раздел 4. Модульное программирование		

3. Варианты оценочных средств для текущего контроля

Вопросы подобраны таким образом, чтобы можно было проверить подготовку студентов по усвоению соответствующих знаний и умений изученной дисциплины.

Предлагается пакет тестовых заданий по оценке качества подготовки студентов. Пакет содержит проверочные тесты, с помощью которых преподаватель может проверить качество усвоения пройденного материала:

- часть А – 80 заданий с кратким ответом – проверка теоретических знаний (задания закрытого типа);
- часть В – комплексный практический тест с 20 заданиями открытого типа;
- часть С – комплексный практический тест с 8 заданиями открытого развернутого типа.

С целью проверки знаний и умений изученной дисциплины каждый студент получает следующий пакет:

Часть А (проверка теоретических знаний) - информационный тест, включающий в себя 20 заданий.

Часть А тестового задания включает в себя:

- выбор правильного ответа;
- множественный выбор;
- установление соответствия;
- установление правильной последовательности;
- закончить предложение.

За каждый правильный ответ – 2 балла.

Максимальное количество баллов – 40.

Часть В (проверка практических знаний и умений) - комплексный практический тест, включающий в себя 8 заданий открытого типа со свободным ответом.

За каждый правильный ответ – 5 баллов.

Максимальное количество баллов – 40.

Часть С (проверка практических знаний и умений) - комплексный практический тест (письменное задание), включающий в себя 2 задания повышенного уровня сложности открытого типа с развернутым ответом.

За каждый правильный ответ – 10 баллов.

Максимальное количество баллов – 20.

ТЕСТОВЫЕ ЗАДАНИЯ

Часть А

1. Язык программирования Паскаль является:

1. языком машинных команд
2. языком высокого уровня
3. языком управления базами данных
4. языком манипулирования данными

2. Какие из терминов означают процесс перевода программ, написанных на языке программирования высокого уровня, в машинные коды:

1. компиляция
2. интерпретация
3. архивация
4. манипуляция

3. Целочисленный тип в Паскале описывается служебным словом:

1. integer

2. real
3. char
4. boolean

4. Какой объем памяти ПК занимает массив A, описанный как
var A: array [1...100] of integer:

1. 100 байт
2. 200 байт
3. 300 байт
4. 400 байт

5. Результат выполнения операции $5 \bmod 2$:

1. 2.5
2. 1
3. 2
4. 5

6. Результат выполнения функции $\text{sqr}(4)$:

1. 2
2. 4
3. 8
4. 16

7. Даны множества $A=[1,2,3]$ и $B=[1..5]$. Какое из высказываний для них справедливо:

1. $A=B$
2. $A \leq B$
3. $A \geq B$
4. $A \langle \rangle B$

8. Типизированные файлы описываются служебным словом:

1. file
2. text
3. record
4. file of тип

9. Оператор ввода в Паскале имеет обозначение:

1. read
2. write
3. input
4. reset

10. Какой из разделов программы является первым:

1. var
2. label
3. type
4. const

11. Автономно компилируемая программная единица – это:

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

12. Каких функций не существует в Паскале:

1. $\cos(x)$

2. $\sin(x)$
3. $\text{tg}(x)$
4. $\text{ctg}(x)$

13. Найдите ошибку:

1. `var c: char; begin c:=chr(100); end;`
2. `var c: char; begin c:=chr(200); end;`
3. `var c: char; begin c:=chr(300); end;`
4. `var c: char; begin c:=chr(255); end;`

14. Какое действие выполняет следующая последовательность операторов:

`for c : ='a' to 'z' do`

`if c IN A then write(c);`

1. ввод элементов множества A
2. вывод элементов множества A
3. нахождение количества элементов множества A
4. нахождение среднего значения

15. Язык программирования Паскаль был создан:

1. Б. Паскалем
2. К. Дж. Дейтом
3. Н. Виртом
4. К.Моучли

16. Какой из видов языков программирования не нуждается в трансляторе:

1. языки машинных команд
2. языки класса автокод-ассемблер
3. языки высокого уровня
4. языки манипулирования данными

17. Какой объем памяти ПК занимает массив B, описанный как

`Var B: array [1...10] of real;`

1. 10 байт
2. 30 байт
3. 80 байт
3. 60 байт

18. Результат выполнения операции $5 \text{ div } 2$:

1. 2.5
2. 1
3. 2
4. 5

19. Результат выполнения функции $\text{sqrt}(4)$:

1. 2
2. 4
3. 8
4. 16

20. Найдите ошибки:

1. `A : set of integer;`
2. `A : set of char;`
3. `A : set of 1..100;`
4. `A : set of real;`

21. Текстовые файлы описываются служебным словом:

1. file
2. text
3. file of тип
4. unit

22. Оператор вывода в Паскале имеет обозначение:

1. read
2. write
3. output
4. reset

23. В каком из разделов программы описываются числа, имеющие громоздкую запись:

1. const
2. var
3. type
4. label

24. Заголовок модуля начинается словом:

1. INTERFACE
2. UNIT
3. IMPLEMENTATION
4. USES

25. В качестве базового типа множества не могут использоваться:

1. вещественный тип
2. интервальный тип
3. логический тип
4. целочисленный тип

26. Какое действие выполняет следующая последовательность операторов:

for i := 1 to 100 do

if i IN A then k:=k+1;

1. ввод элементов множества A
2. вывод элементов множества A в алфавитном порядке
3. нахождение количества элементов множества A
4. нахождение произведения элементов множества A

27. Расставьте языки программирования высокого уровня в порядке их возникновения:

1. Алгол
2. Бейсик
3. Паскаль
4. Си

Ответ: 1,2,3,4

28. В каком из языков программирования высокого уровня трансляция реализуется методом интерпретации:

1. Паскаль
2. Бейсик
3. Си
4. Фортран

29. Результат выполнения функции odd(3):
1. false
 2. true
 3. 0
 4. 1
30. Результат выполнения функции abs(-5):
1. 5
 2. 25
 3. -25
 4. 0
31. В качестве базового типа множества не может использоваться:
1. вещественный тип
 2. интервальный тип
 3. логический тип
 4. символьный тип
32. Нетипизированные файлы описываются служебным словом:
1. file
 2. text
 3. file of тип
 4. procedure
33. Какая из последовательностей символов не может служить идентификатором переменной:
1. x1
 2. 1x
 3. time
 4. AAA
34. Какой раздел программы необходим для применения оператора безусловного перехода goto:
1. var
 2. type
 3. label
 4. const
35. Интерфейсная часть модуля начинается со слова:
1. INTERFACE
 2. UNIT
 3. IMPLEMENTATION
 4. USES
36. Комбинированный тип описывается:
1. в разделе переменных
 2. в разделе типов
 3. в разделе констант
 4. в разделе перменных
37. Расставьте языки программирования высокого уровня в порядке их возникновения:
1. Фортран
 2. Бейсик
 3. Паскаль
 4. Си

Ответ 1,2,3,4

38. В Турбо-Паскале компиляция программы осуществляется по нажатию клавиши:

1. F1
2. F5
3. F7
3. F9

39. Символьный тип в Паскале имеет область значений состоящую из:

1. 64 символов
2. 128 символов
3. 256 символов
4. 512 символов

40. Результат выполнения функции round(4.8):

1. 5
2. 4
3. 0
4. 48

41. Процедура ASSIGN (f, имя файла) применяется для:

1. связывания файловой переменной с именем файла
2. открытия файла для чтения
3. открытия файла для записи
4. закрытия файла

42. Результат выполнения функции ord(c) не может принимать значение:

1. 244
2. 128
3. 302
4. 255

43. В каком разделе программы описываются переменные:

1. type
2. var
3. label
4. const

44. Исполняемая часть модуля начинается со слова:

1. INTERFACE
2. UNIT
3. IMPLEMENTATION
4. USES

45. Какая из операций всегда возвращает вещественный результат:

1. +
2. *
3. /
4. -

46. Комбинированный тип описывается служебным словом:

1. record
2. array
3. set of
4. type

47. Какой из языков программирования высокого уровня предназначен для разработки системных и прикладных программ:

1. Паскаль
2. Си
3. Бейсик
4. Фортран

48. В Турбо-Паскале запуск программы осуществляется по нажатию комбинации клавиш:

1. Alt-F5
2. Ctrl-F9
3. Alt-X
4. Ctrl-F5

49. Логический тип в Паскале имеет область значения, состоящую из:

1. одного значения
2. двух значений
3. четырех значений
4. 256 значений

50. Результат выполнения функции $\text{abs}(x)$ имеет (в этом вопросе несколько правильных ответов):

1. вещественный результат при любом x
2. вещественный результат при целочисленном x
3. вещественный результат при вещественном x
4. целочисленный результат при целочисленном x

51. Результат выполнения функции $\text{trunc}(4.8)$:

1. 4
2. 5
3. 0
4. 48

52. Процедура RESET (f) применяется для:

1. связывания файловой переменной с именем файла
2. открытия файла для чтения
3. открытия файла для записи
4. закрытия файла

53. Просмотр результатов выполнения программы осуществляется по нажатию комбинации клавиш:

1. Alt-F5
2. Ctrl-F9
3. Alt-X
4. F9

54. Объявления всех глобальных объектов модуля содержит:

1. исполняемая часть
2. интерфейсная часть
3. иницилирующая часть
4. заголовок модуля

55. Каким типом должна быть описана переменная, являющаяся индексом массива:

1. integer
2. real
3. char

4. boolean

56. Какой из языков программирования высокого уровня первым претендовал на универсальность:

1. ПЛ/1
2. Паскаль
3. Си
4. Алгол

57. Выход из системы Турбо-Паскаль осуществляется по нажатию комбинации клавиш:

1. Alt-F5
2. Ctrl-F9
3. Alt-X
4. F9

58. Какие из перечисленных ниже типов данных Паскаля занимают больше одного байта памяти ПК:

1. boolean
2. char
3. integer
4. real

59. Какое описание является аналогом следующего C: string[10]:

1. C: array[1..10] of char
2. C: array[1..10] of integer
3. C: array[1..10] of real
4. C: array[1..10] of boolean

60. Для определения равенства двух множеств, какие из условий необходимы:

1. оба множества имеют одинаковое количество элементов
2. оба множества содержат одинаковые элементы
3. оба множества имеют одинаковые названия
4. оба множества имеют одинаковую мощность

61. Процедура REWRITE (f) применяется для:

1. связывания файловой переменной с именем файла
2. открытия файла для чтения
3. открытия файла для записи
4. закрытия файла

62. Оператор присваивания в Паскале имеет обозначение:

1. =
2. :=
3. ==
4. <>

63. Распределите разделы программы в правильном порядке:

1. label
2. const
3. type
4. var

Ответ 1,2,3,4

64. Установите соответствие:

Тип данных		Его описание	
1.	целочисленный	А	integer
2.	вещественный	Б	real
3.	символьный	В	char
4.	логический	Г	boolean

Ответ 1 А, 2 Б, 3 В, 4 Г

65. Глобальные переменные процедур и функций описываются:

1. в теле подпрограммы
2. в заголовке подпрограммы
3. в теле программы
4. в заголовке программы

66. Локальные переменные процедур и функций описываются:

1. в теле подпрограммы
2. в заголовке подпрограммы
3. в теле программы
4. в заголовке программы

67. Какое описание является аналогом String[10]

1. Array [1..10] of char
2. Set of 1..10
3. Array [1..5, 1..5]
4. Record

68. Процедура Length(st)

1. вычисляет длину строки в байтах
2. преобразует значение строки в число
3. удаляет символ из строки
4. вставляет символ в строку

69. Процедура Val (St, Ch, Code)

1. вычисляет длину строки в байтах
2. преобразует значение строки в число
3. удаляет символ из строки
4. вставляет символ в строку

70. Какой из видов языков программирования являются машинно-ориентированными:

1. языки машинных команд
2. языки класса автокод-ассемблер
3. языки высокого уровня
4. языки манипулирования данными

71. Расставьте в правильном порядке этапы решения задач на ЭВМ

1. этап разработки математической модели решаемой задачи
2. этап разработки методики решения и определения ограничений на решаемую задачу
3. этап разработки алгоритма решаемой задачи
4. этап программирования решаемой задачи на одном из языков программирования
5. этап тестирования и отладки программы

Ответ 1,2,3,4,5

72. Какая форма описания алгоритма отсутствует?

1. словесная
2. блок-схема
3. запись на языке программирования
4. структурная схема

73. Последовательность действий, направленных на получение определённого результата за конечное число шагов - это

1. алгоритм
2. программа
3. программный цикл
4. программный продукт

74. Что из перечисленного не является логическим высказыванием?

1. 6 – четное число
2. Информатика – интересный предмет
3. Ученик десятого класса
4. В городе А более миллиона жителей

75. Какому десятичному числу соответствует двоичное число 0111011?

1. 57
2. 58
3. 59
4. 60

76. Чему равно двойное отрицание X?

1. X
2. \bar{X}
3. 0
4. 1

77. Чему равно $X \cup 0$?

1. X
2. \bar{X}
3. 1
4. 0

78. Чему равно $X \cap \bar{X}$?

1. 1
2. \bar{X}
3. X
4. 0

79. Какому двоичному числу соответствует десятичное число 133?

1. 1000101
2. 1010101
3. 1001001
4. 1101101

80. Конъюнкция – это

1. логическое сложение
2. логическое умножение
3. логическое деление
4. логическое вычитание

Часть В

1. Какое значение будет иметь переменная d для заданных a=5, b=3, c=1:
if (a<b) and (b<c) then d:= c-a+b
else
if (a>b) and (b>c) then d:= a-b+c
else
if (a>b) and (b<c) then d:= a-c+b;

Ответ:

2. Найти значение c при a=3,b=5,c=1:

```
while a<b do
  begin
    c:= c+a*b;
    a:= a+1;
  end;
```

Ответ:

3. Найти значение c при a=10, b=5:

```
repeat
  b:= b+1;
  a:= a-b;
  c:= a+b;
until a≤0;
```

Ответ:

4. Найти значение c для заданных a=3,b=2,c=4,n=4:

```
for i:=1 to n do
  c:= c+a*b;
```

Ответ:

5. Определить, что находит программа:

```
var A: array [1..10] of integer; S, i: integer;
begin
  S:= A[1];
  for i:= 2 to 10 do
    if S>A[i] then S:=A [i];
end.
```

Ответ:

6. Найти значение S:

```
for i:=1 to 6 do
  S:=0;
begin
  A [i]:=i;
  S:=S+A [i];
end;
```

Ответ:

7. Найти значение S:

```
for i:=1 to 4 do
  begin
    A [i]:=i;
    S:=S*A [i];
  end;
```

S:=S/i;

Ответ:

8. Найти значение S:

S: = 1;

for i: =1 to 5 do

begin

A [i]:=i;

S:=S*A [i];

end;

Ответ:

9. Найти значение d при a=3:

var a, b, c: integer; d: real;

begin

b: = a*a;

c: = a+b;

d: = c/a;

end.

Ответ:

10. Найти значение c при a=2, b=4, c=3:

while a<b do

begin

c:= c+a*b;

a:= a+1;

end;

Ответ:

11. Найти значение c при a=9, b=4:

repeat

b:= b+1;

a:= a-b;

c:= a+b;

until a≤0;

Ответ:

12. Найти значение c для заданных a=2, b=4, c=3, n=3:

for i:=1 to n do

c:= c+a*b;

Ответ:

13. Определить, что находит программа:

var A: array [1..10] of integer; S,I: integer; SA: real;

begin

S:=0;

for i:=1 to n do

S:=S+A[i];

SA:=S/n;

end;

Ответ:

14. Найти значение S:

for i:=1 to 5 do

begin

```
A [i]:=i;  
S:=S*A [i];  
end;
```

Ответ:

15. Какое значение будет иметь переменная d для заданных a=2, v=3, c=4:

```
if (a<b) and (b<c) then d:= c-a+b  
else  
  if (a>b) and (b>c) then d:= a-b+c  
  else  
    if (a>b) and (b<c) then d:= a-c+b;
```

Ответ:

16. Найти значение c при a=3, v=5, c=2:

```
while a<b do  
begin  
  c:= c+a*b;  
  a:= a+1;  
end;
```

Ответ:

17. Найти значение c при a=8, b=3:

```
repeat  
  b:= b+1;  
  a:= a-b;  
  c:= a+b;  
until a≤0;
```

Ответ:

18. Найти значение c для заданных a=3, b=2, c=5, n=3:

```
for i:=1 to n do  
  c:= c+a*b;
```

Ответ:

19. Определить, что находит программа:

```
var A: array [1..10] of integer; S,I: integer;  
begin S:=0;  
  for i:=1 to n do  
    S:=S+A[i];  
end.
```

Ответ:

20. Найти значение S:

```
for i:=1 to 6 do  
begin  
  A [i]:=I;  
  S:=S+A [i];  
end;  
S:=S/I;
```

Ответ:

Часть С

1. Подсчитать сумму чисел кратных 5 до первого нулевого значения.
(Использовать оператор цикла с предусловием или с постусловием).
2. Дан массив из 100 целых чисел. Написать программу нахождения суммы положительных элементов массива, не превосходящих заданного числа А.
3. В целочисленной матрице размерностью 10*10 найти наибольший элемент и номер строки, в которой он находится
4. Подсчитать количество нечетных чисел до первого нулевого значения.
(Использовать оператор цикла с предусловием или с постусловием).
5. Дан массив из 100 целых чисел. Написать программу нахождения произведения положительных элементов массива, не превосходящих заданного числа А.
6. В целочисленной матрице размерностью 10*10 найти наименьший элемент и номер столбца, в котором он находится.
7. Подсчитать сумму нечетных чисел до первого нулевого значения.
(Использовать оператор цикла с предусловием или с постусловием).
8. Дан массив из 100 целых чисел. Написать программу нахождения произведения отрицательных элементов массива с четными индексами.

КРИТЕРИИ ПО ВЫСТАВЛЕНИЮ БАЛЛОВ

Сводная таблица с критериями баллов	
Части	Баллы
А	40
В	40
С	20
Итого (макс. баллы)	100

Критерии оценок	
Баллы	Оценки
86-100	5
71-85	4
49-70	3
Менее 48 баллов	перезачет

контрольная работа по дисциплине «основы алгоритмизации и программирования»

1. Программа на С++ состоит:
 - Из операторов отделенных точкой запятой
 - Из операторов заключенных в фигурные скобки
 - Как минимум из одной функции main()
2. модификатор const примененный к переменной базового типа
 - сообщает компилятору что переменная продолжает существовать в течение всего времени выполнения программы.
 - Предотвращает непосредственное изменение таких переменных в программе.

3. переменные, объявленные вне блоков программы имеют
 - глобальную область видимости
 - область видимости блока
4. цикл do-while
 - позволяет выполнять операторы, указанные в его теле, определенное количество раз
 - позволяет выполнять операторы до тех пор, пока его проверочное условие возвращает true
 - выполняет операторы как минимум один раз, а затем позволяет повторять их до тех пор. Пока проверочное условие истинно.
5. операция логического «или» (| |), применяется тогда,
 - когда есть два условия, и оба должны вернуть результат true, что бы общий результат был true
 - когда есть два условия и нужно получить true, если любое из них или оба возвращают true
6. указатель это –
 - псевдоним другой переменной, который может использоваться в тех же местах что и переменная
 - переменная, содержащая адрес другой переменной.
7. К какому типу относится контейнер vector?
 - Последовательный
 - Ассоциативный
8. Контейнеры map, multimap, hashmap относятся к типу
 - последовательных
 - ассоциативных
9. Контейнер map позволяет хранить пары значений, где ключи всегда
 - уникальные
 - любые
10. Итератор -это
 - указатель на позицию в контейнере
 - функция - возвращающая позицию выбранного элемента
 - массив значений контейнера
11. функция класса vector begin() возвращает
 - значение первого элемента контейнера
 - итератор на начало контейнера

контрольная работа по дисциплине «основы алгоритмизации и программирования»

1. Программа на C++ состоит:
 - Из операторов отделенных точкой запятой
 - Из операторов заключенных в фигурные скобки
 - Как минимум из одной функции main()
2. модификатор const примененный к переменной базового типа
 - a. сообщает компилятору что переменная продолжает существовать в течение всего времени выполнения программы.
 - b. Предотвращает непосредственное изменение таких переменных в программе.
3. переменные, объявленные вне блоков программы имеют

- a. глобальную область видимости
 - b. область видимости блока
4. цикл do-while
- a. позволяет выполнять операторы, указанные в его теле, определенное количество раз
 - b. позволяет выполнять операторы до тех пор, пока его проверочное условие возвращает true
 - c. выполняет операторы как минимум один раз, а затем позволяет повторять их до тех пор. Пока проверочное условие истинно.
5. операция логического «или» (||), применяется тогда,
- a. когда есть два условия, и оба должны вернуть результат true, что бы общий результат был true
 - b. когда есть два условия и нужно получить true, если любое из них или оба возвращают true
6. указатель это –
- a. псевдоним другой переменной, который может использоваться в тех же местах что и переменная
 - b. переменная, содержащая адрес другой переменной.
7. К какому типу относится контейнер vector?
- Последовательный
 - Ассоциативный
8. Контейнеры map, multimap, hashmap относятся к типу
- последовательных
 - ассоциативных
9. Контейнер map позволяет хранить пары значений, где ключи всегда
- уникальные
 - любые
10. Итератор -это
- указатель на позицию в контейнере
 - функция - возвращающая позицию выбранного элемента
 - массив значений контейнера
11. функция класса vector begin() возвращает
- значение первого элемента контейнера
 - итератор на начало контейнера

- 21.** Тело цикла - это...
- а) группа команд, не входящих в циклическую структуру;
 - б) группа команд, повторяющихся некоторое число раз;
 - в) произвольная группа команд;
 - г) команды, заключенные в операторные скобки.
- 22.** Какой из перечисленных заголовков циклов не содержит ошибок?
- а) FOR X:=3 TO 12 DO;
 - б) FOR K = 1 TO 5.5 DO;
 - в) FOR K = 2DOWNTO 10 DO;
 - г) FOR J := 10 TO -2 DO
- 23.** FOR J := 10 TO -2 DO
- 24.** Что определяет индекс массива?
- а) Индекс определяет положение элемента массива данных относительно его конца;
 - б) Индекс определяет положение элемента массива данных относительно друг друга;
 - в) Индекс определяет положение элемента массива данных относительно его начала.
- 25.** Укажите правильное описание массива
- а) Var a:array[1..1000] of integer;
 - б) Var A: ARRAY [1..50 OF REAL];
 - в) Var A, B, C: ARRAY [1 ..50] OF REAL.
- 26.** Что производит следующий фрагмент программы?
- ```
s:=0; for i:=1 to n do
for j:=1 to n do s: =s+a[i, j]; end;
writeln('s=',s);
```
- а) Находит сумму элементов каждой строки и выводит их на экран;
  - б) Находит сумму всех элементов двумерного массива и сумму элементов каждой строки и выводит их на экран;
  - в) Находит сумму всех элементов двумерного массива и выводит их на экран.
- 27.** Что определяет индекс массива?
- г) Индекс определяет положение элемента массива данных относительно его конца;
  - д) Индекс определяет положение элемента массива данных относительно друг друга;
  - е) Индекс определяет положение элемента массива данных относительно его начала.
- 28.** Укажите правильное описание массива
- г) Var a:array[1..1000] of integer;
  - д) Var A: ARRAY [1..50 OF REAL];
  - е) Var A, B, C: ARRAY [1 ..50] OF REAL.
- 29.** Что производит следующий фрагмент программы?
- ```
s:=0; for i:=1 to n do
for j:=1 to n do s: =s+a[i, j]; end;
writeln('s=',s);
```
- г) Находит сумму элементов каждой строки и выводит их на экран;
 - д) Находит сумму всех элементов двумерного массива и сумму элементов каждой строки и выводит их на экран;
 - е) Находит сумму всех элементов двумерного массива и выводит их на экран.

ВОПРОСЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ- ДИФФЕРЕНЦИРОВАННЫЙ ЗАЧЁТ

Вопросы

1. Алгоритм. Понятие. Определение. Свойства.
2. Способы описания алгоритмов. Правила описания схем алгоритмов.
3. Разновидности структур алгоритмов.
4. Лексемы языка C/C++. Идентификаторы. Операции.
5. Лексемы языка C/C++. Ключевые слова. Константы.
6. Переменные. Описание переменных.
7. Переменные. Типы данных в C/C++.
8. Структура программы. Команда присваивания в C/C++.
9. Ввод-вывод данных. Функции scanf, printf. Формат выводимых данных.
10. Ввод-вывод данных. Стандартные потоки ввода и вывода. Примеры.
11. Алгоритм линейной структуры.
12. Операции отношений. Логические операции в C. Таблицы истинности.
13. Поразрядные операции в C. Примеры.
14. Структура IF(если- то- иначе) и ее программирование. Примеры.
15. Структура switch(выбор) и ее программирование. Примеры.
16. Оператор безусловного перехода GOTO. Примеры.
17. Алгоритмы циклической итерационной структуры. Оператор цикла While. Примеры использования.
18. Алгоритмы циклической итерационной структуры. Оператор цикла do... while. Примеры использования.
19. Алгоритмы циклической итерационной структуры. Оператор цикла For... . Примеры использования.
20. Операторы break и continue. Примеры использования.
21. Одномерные массивы. Задание массивам первоначальных значений.
22. Операции над массивами и их совместимость. Ввод-вывод массивов.
23. Ввод-вывод матриц. Операции над матрицами.
24. Квадратная матрица. Диагонали матрицы. Пример алгоритма обработки квадратных матриц.
25. Нахождение максимального (минимального) элемента массива.
26. Понятие подпрограммы. Описание подпрограммы. Протопит функции.
27. Понятие о стандартных директивах препроцессора.
28. Формальные и фактические параметры.
29. Область видимости имен.
30. Понятие о локальных и глобальных переменных.
31. Рекурсия. Виды рекурсий.
32. Обработка строковых данных. Операции со строками.
33. Понятие структуры. Массивы структур. Обработка структур.
34. Файлы. Типы файлов. Способы их описания.
35. Способы доступа к файлам.
36. Операции ввода/вывода с файлами через буфер.
37. Текстовые файлы. Стандартные текстовые файла.
38. Двоичные файлы. Функции ввода и вывода данных.

39. Использование файлового указателя при программировании операций с файлами.
40. Стандартные функции для работы с файлами в стиле C.
41. Работа с файлами в стиле C++. Запись в файл. Чтение файлов.
42. Перечисляемые типы данных.
43. Интегрированная среда разработки C++**Builder**. Интерфейс среды. Характеристика проекта C++**Builder**.
44. Основные компоненты панели Standard. Компиляция и выполнение проекта.
45. Сортировка методом простого выбора. Суть. Алгоритм. Эффективность.
46. Сортировка методом простого обмена. Суть. Алгоритм. Эффективность.
47. Сортировка методом прямого включения. Суть. Алгоритм. Эффективность.
48. Быстрая сортировка. Суть. Алгоритм. Эффективность.
49. Модификация метода Прямого включения. Сортировка Шелла. Суть. Алгоритм. Эффективность.
50. Модификация метода пузырька. Шейкерная сортировка. Суть. Алгоритм. Эффективность.
51. Карманная (корзинная) сортировка. Суть. Алгоритм. Эффективность.
52. Алгоритмы поиска информации. Линейный поиск. Суть. Алгоритм. Эффективность.
53. Алгоритмы поиска информации. Бинарный поиск. Суть. Алгоритм. Эффективность.
54. Понятие рекурсии и основные определения. Примеры.
Общие сведения о динамическом распределении памяти и динамических переменных.
55. Указатели, объявление ссылочных переменных.
56. Управление выделением и освобождением динамической памяти на логическом уровне.
57. Создание однонаправленного списка. Добавление элемента в список. Примеры.
58. Создание однонаправленного списка. Удаление элемента из списка. Примеры.
59. Создание двунаправленного списка. Добавление и удаление элемента из начала списка. Примеры.
60. Создание двунаправленного списка. Удаление элемента из списка по ключу. Примеры.
61. Понятие стека. Примеры использования стека в программировании.
62. Основные приемы и особенности работы со стеками. Добавление элемента в стек. Примеры.
63. Основные приемы и особенности работы со стеками. Удаление элемента из стека. Примеры.
64. Способы организации очереди. Примеры.
65. Обработка очередей. Добавление элемента в очередь. Примеры.
66. Обработка очередей. Удаление элемента из очереди. Примеры.
67. Понятие Дерево. Бинарное дерево поиска.
68. Бинарное Дерево. Формирование Дерева. Добавление узла в Дерево поиска.
69. Бинарное Дерево. Рекурсивный и нерекурсивный поиск узла в Дереве.
70. Бинарное Дерево. Алгоритмы Обхода Дерева.
71. Бинарное Дерево. Удаление узла из Дерева.
72. Обработка Бинарного Дерева. Подсчет узлов дерева.
73. Обработка Бинарного Дерева. Определение высоты дерева.

Критерии оценивания на экзамене

- Оценка **«отлично»** выставляется обучающемуся, если демонстрируются: всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

- Оценка **«хорошо»** выставляется обучающемуся, если демонстрируются: достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

- Оценка **«удовлетворительно»** выставляется обучающемуся, если демонстрируются: заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

- Оценка **«неудовлетворительно»** выставляется обучающемуся, если обнаруживаются пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнившего самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по дисциплине.

