Министерство образования Иркутской области
Государственное бюджетное профессиональное образовательное учреждение Иркутской области
«Иркутский авиационный техникум»

УТВЕРЖДАЮ
И.О. директора
ГБПОУИО «ИАТ»
_____ Якубовский А.Н.
«31» мая 2017 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

ОП.08 Теория алгоритмов

специальности

09.02.03 Программирование в компьютерных системах

Иркутск, 2017

Рассмотрена
цикловой комиссией
ПКС протокол № 12 от
19.05.2017 г.

Председатель ЦК

 /М.А. Кудрявцева /

№	Разработчик ФИО
1	Еримеев Антон Михайлович

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.1. Область применения фонда оценочных средств (ФОС)

ФОС по дисциплине является частью программы подготовки специалистов среднего звена по специальности 09.02.03 Программирование в компьютерных системах

1.2. Место дисциплины в структуре ППССЗ:

ОП.00 Общепрофессиональный цикл.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины

В результате освоения дисциплины обучающийся должен	№ дидактической единицы	Формируемая дидактическая единица
Знать	1.1	основные модели алгоритмов;
	1.2	методы построения алгоритмов;
	1.3	методы вычисления сложности работы алгоритмов
Уметь	2.1	разрабатывать алгоритмы для конкретных задач;
	2.2	определять сложность работы алгоритмов;

1.4. Формируемые компетенции:

ОК.1 Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.
ОК.2 Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.
ОК.3 Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.
ОК.4 Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.
ОК.5 Использовать информационно-коммуникационные технологии в профессиональной деятельности.
ОК.6 Работать в коллективе и в команде, эффективно общаться с коллегами,

руководством, потребителями.
ОК.7 Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.
ОК.8 Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.
ОК.9 Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

ПК.1.1 Выполнять разработку спецификаций отдельных компонент.
ПК.1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

2. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

2.1 Текущий контроль (ТК) № 1

Тема занятия: 1.1.4.Оценка эффективности алгоритма.
Метод и форма контроля: Тестирование (Опрос)
Вид контроля: Компьютерное тестирование
Дидактическая единица: 1.1 основные модели алгоритмов;
Занятие(-я):
1.1.1.Введение в дисциплину. Понятие алгоритма. Типы алгоритмов, свойства.
1.1.2.Способы описания алгоритмов. Блок-схема

Задание №1

Решите тест из 20 вопросов.
Каждый правильный ответ на вопрос оценивается в один балл. Возможен только один правильный ответ в каждом вопросе. Максимальное количество баллов - 20.

1. Что такое алгоритм?
   набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий.
   инсталляция порядка действий для ЭВМ, который составлен для определенного языка программирования и удовлетворяет определенным правилам.
   инсталляция порядка действий для ЭВМ, который составлен и удовлетворяет определенным правилам
   искусство счета с помощью цифр, но поначалу слово «цифра» относилось только к нулю

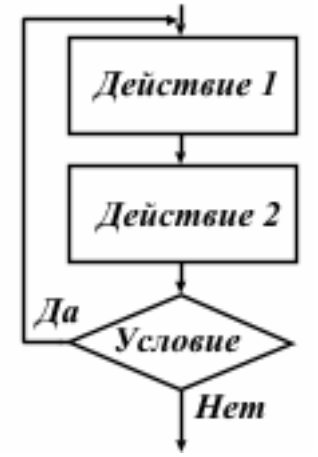
2. Какое из перечисленных свойств не относится к свойствам алгоритмов?
   Дискретность Определенность Результативность Полнота Массовость Правильность

3. Какое свойство алгоритмов характеризует данное определение: «Алгоритм должен приводить к решению за конечное число шагов»
   Дискретность Определенность Результативность Полнота Массовость Правильность

4. Какое свойство алгоритмов характеризует данное определение: «Каждое правило алгоритма должно быть четким, однозначным»
   Дискретность Определенность Результативность Полнота Массовость Правильность

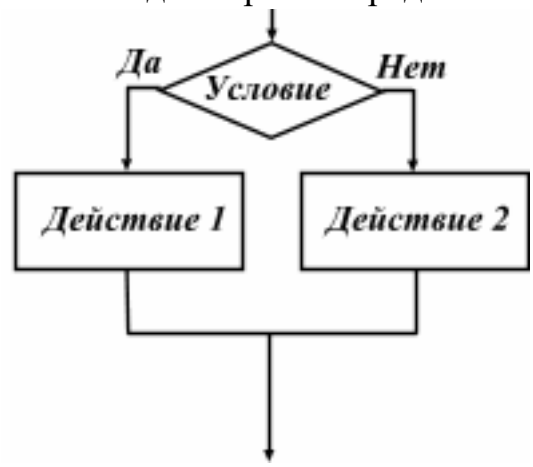
5. Какое свойство алгоритмов характеризует данное определение: «Алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными»
   Дискретность Определенность Результативность Полнота Массовость Правильность

6. Какой вид алгоритма представлен на рисунке?



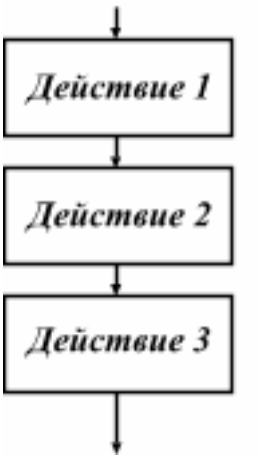
   Линейный Разветвляющий Циклический

7. Какой вид алгоритма представлен на рисунке?



   Линейный Разветвляющий Циклический

8. Какой вид алгоритма представлен на рисунке?



Линейный Разветвляющий Циклический

9. Выберите вариант, где представлены все способы описания алгоритма (но нет лишних).
   Словесный, Формульно-словесный, Графический, Программный
   Словесный, Формульный, Формульно-словесный, Графический, Программный
   Словесный, Формульный, Графический, Программный
   Формульный, Графический, Программный

10. Что такое блок схема?
    распространенный тип схем, описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.
    схематичное изображение элементов алгоритма, которые собранны воедино с помощью линий в определенном порядке, демонстрируя алгоритм.
    строгое изображение элементов алгоритма, которые собранны воедино по правилам с помощью линий в определенном порядке, демонстрируя алгоритм.
    схема программы на языке программирования, представленная в виде блоков, соединенных линиями

11. Блок схема. На рисункепредставлено обозначение элемента алгоритма. Выберите его наименование



    Блок начало-конец
    Блок действия
    Логический блок (блок условия)
    Данные

12. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

13. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

14. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

15. В чем заключается Разработка алгоритма решения задачи
в установлении необходимой последовательности арифметических и логических действий, строгое выполнение которых приводит к решению задачи.
строгое выполнение арифметических и логических задач в необходимой последовательности в программном коде
выполнение арифметических и логических задач в необходимой последовательности в программном коде несет рекомендательный характер

16. Вставьте пропущенное слово: « Любой алгоритм применяется к … и выдает результат»
    исходным данным
    внешним данным
    используемым средствам
    внутренним средствам

17. Вставьте пропущенное слово: «Данные для своего размещения требуют …. … обычно считается дискретной. Единицы измерения … и данных должны быть согласованы между собой»
    память
    элементарный шаг
    участок
    промежуток

18. Вставьте пропущенное слово: «Алгоритм состоит из отдельных …»
    Участков памяти
    элементарный шагов
    участков
    промежутков

19. Вставьте пропущенное слово: «Последовательность шагов алгоритма …, т.е. после каждого шага указывается следующий шаг, либо алгоритм останавливается»
    детерминирована
    результативна
    объективна
    полна

20. Вставьте пропущенное слово: «Каждый алгоритм должен быть … , т.е. после конечного числа шагов выдавать результат»
    детерминировным
    результативным
    объективным
    полным

Оценка	***Показатели оценки***
3	Обучающийся ответил на 6-13 вопросов
4	Обучающийся ответил на 14-17 вопросов

5	Обучающийся ответил на 18-20 вопросов

2.2 Текущий контроль (ТК) № 2

Тема занятия: 2.1.5.Решение задач на составление разветвляющихся алгоритмов
Метод и форма контроля: Письменный опрос (Опрос)
Вид контроля: Проверочная работа
Дидактическая единица: 1.2 методы построения алгоритмов;
Занятие(-я):
1.1.2.Способы описания алгоритмов. Блок-схема
1.1.3.Основы Pascal. Операторы, функции, процедуры.
2.1.1.Линейные алгоритмы
2.1.3.Разветвляющие алгоритмы

Задание №1

Выполнить 5 теоретических заданий (в первых двух заданиях ответ - действительные числа, в трех последующих - выбор ответа).
Каждое правильно выполненное задание оценивается в один балл. Возможен только один правильный ответ в каждом задании. Максимальное количество баллов - 5.

1. Определите значение переменной c после выполнения следующего фрагмента программы

```
a := 60;

b := 20;

a := a / b * 2;

if a > b then

  c := a - 4 * b

else

  c := a + 4 * b;
```

2. Определите значение переменной c после выполнения следующего

```
a := 50;
b := 10;
a := a / b * 2;
if a > b then
  c := a - 4 * b
else
  c := a + 4 * b;
```

фрагмента программы

3. Выберите верное утверждение
   Переменная состоит из имени и выделенной области памяти, которая ему соответствует
   Числа являются частью переменных
   После присваивания переменная запоминает число
   Переменные в области памяти задаются независимо от средств вычисления и необходимостей автора. Невозможно изменить нахождения переменных в области памяти

4. Задача: даны две цифры, необходимо узнать какая из них больше и вывести ответ на экран. Какого вида алгоритма будет достаточно для решения данной задачи?
   Линейного
   Разветвляющего
   Циклического

5. В Pascal неполный разветвляющийся алгоритм пишется:
   a:=a/b;
   if (a>b) then a:=b;
   if (a>b) then a:=b else a:=c;

Оценка	***Показатели оценки***
3	Обучающийся выполнил 2-3 задания
4	Обучающийся выполнил 4 задания
5	Обучающийся выполнил 5 заданий

Дидактическая единица: 2.1 разрабатывать алгоритмы для конкретных задач;

Занятие(-я):

2.1.2.Решение задач на составление линейных алгоритмов

2.1.4.Решение задач на составление разветвляющихся алгоритмов

Задание №1

Разработать алгоритм работы программы в виде блок-схемы и программный код на любом языке программирования для следующей задачи:
Дано натуральное число. Определить четное оно или нечетное.

Оценка	***Показатели оценки***
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными.
4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию четных и нечетных чисел. Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»; Блок-схема соответствует выбранному алгоритму решения задачи.

2.3 Текущий контроль (ТК) № 3

Тема занятия: 2.2.6.Решение задач на составление циклических алгоритмов. Вложенные циклы. Определение сложности циклических алгоритмов

Метод и форма контроля: Практическая работа (Опрос)

Вид контроля: Проверочная работа

Дидактическая единица: 1.2 методы построения алгоритмов;

Занятие(-я):

2.2.1.Циклические алгоритмы. Цикл с параметром
2.2.2.Цикл с постусловием и с предусловием
2.2.5.Вложенные циклы. Вычисление сложности циклического алгоритма

Задание №1

Выполните 17 теоретических заданий (в трех первых заданиях ответ - действительные числа, в остальных 14 - выбор ответа).
Каждое правильно выполненное задание оценивается от 1 до 3 баллов (количество баллов указано в формулировке каждого задания). Возможен только один правильный ответ в каждом задании. Максимальное количество баллов - 22.

1. Определите, что будет напечатано в результате выполнения программы (2 балла)

```
var n, s: integer;
begin
  n := 0;
  s := 512;
  while s >= 0 do
  begin
    s := s - 20;
    n := n + 1
  end;
  write(n)
end.
```

2. Получив на вход число x, этот алгоритм печатает два числа *a* и *b*. Укажите НАИБОЛЬШЕЕ из таких чисел x, при вводе которых алгоритм печатает сначала 2, а потом 8 (3 балла)

```
var x, a, b: integer;
begin
  readln(x);
  a:=0; b:=0;
  while x>0 do
  begin
    a:=a+1;
    b:=b+(x mod 10);
    x:=x div 10;
  end;
  writeln(a); write(b);
end.
```

3. Определите, что будет напечатано в результате выполнения программы (2 балла)

```
var n, s: integer;
begin
  n := 24;
  s := 0;
  while n <= 28 do
  begin
    s := s + 20;
    n := n + 2
  end;
  write(s)
end.
```

4. Задача: даны две цифры, необходимо узнать какая из них больше и вывести ответ на экран. Какого вида алгоритма будет достаточно для решения данной задачи? (1 балл)

Линейного
Разветвляющего
Циклического

5. Что такое Цикл (1 балл)
   - это последовательность операторов, которая может выполняться один или более раз
   - это последовательность переменных, которые выполняются один или более раз
   - это последовательность переменных, которые выполняются хаотичное число раз
   - это последовательность операторов, которая может выполняться хаотичное раз

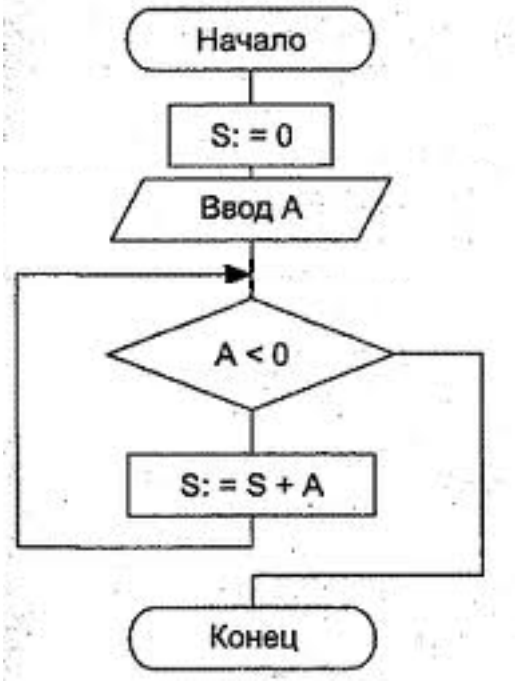
6. К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (while)» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

7. К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (repeat)» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

8. К какому виду цикла относится данное определение: «используется тогда, когда заранее определенное число повторений» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

9. К какому виду цикла относится данное определение: «находится в теле другого цикла» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

10. На Картинке представлена блок-схема. Выберите код данной блок-схемы. (1 балл)




```
Program;
var S;A:=integer;
begin
S:=0;
readln(A);
while A<0 do
begin
S:=S+A;
end;
end.
```

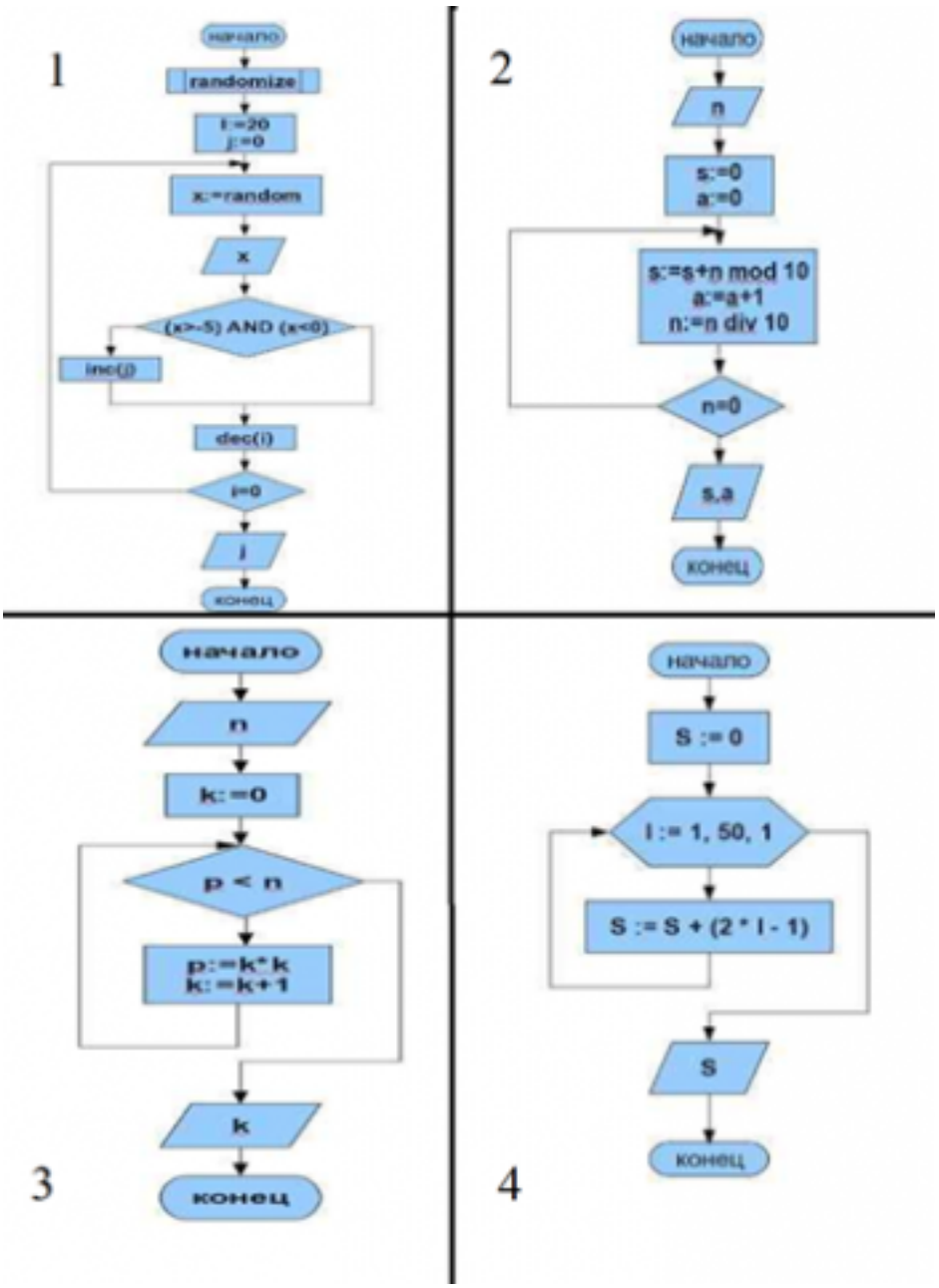
```
Program;
var S;A:=integer;
begin
readln(A);
S:=0;
while A<0 do
begin
S:=S+A;
end;
end.
```

```
Program;
var S;A:=integer;
begin
S:=0;
readln(A);
repeat
S:=S+A; until A<0
end;
end.
```

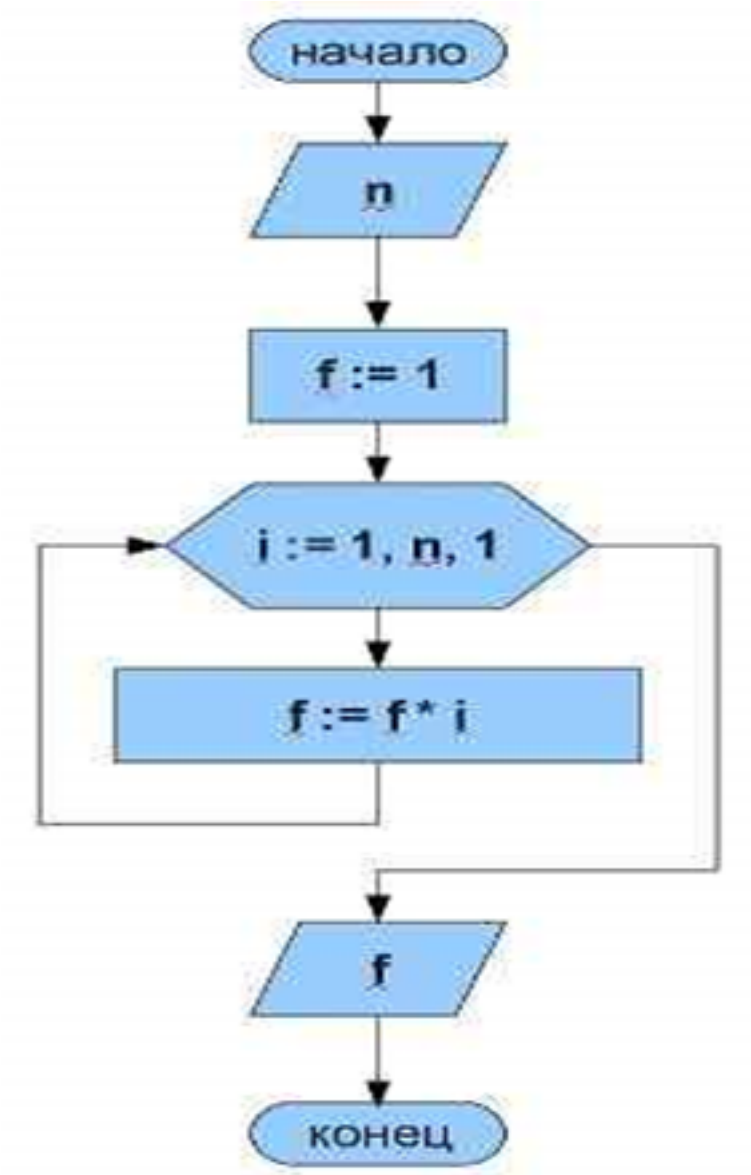
```
Program;
var S;A:=integer;
begin
readln(A);
S:=0;
repeat
S:=S+A; until A<0
end;
end.
```

11. На картинке представлены блок-схемы, выберете нужную по данной программе: (2 балла)

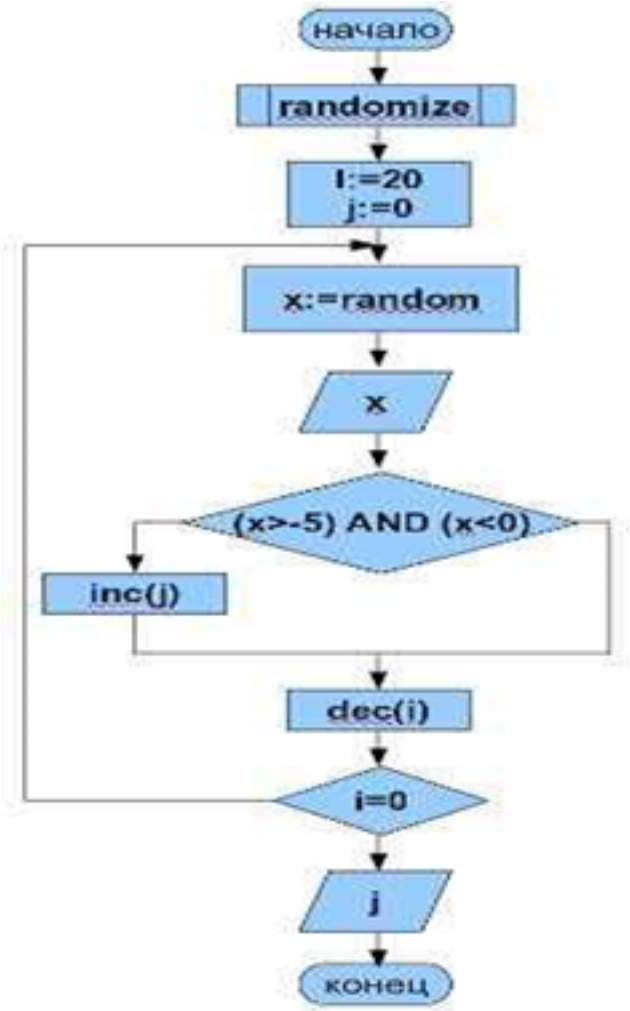
```
program ;
Var
  n,k,p:longint;
Begin
Write(‘Введите число N > 0, N = '); readln(n);
k:=0;
while p < n Do
Begin
  p := k * k;
  k := k + 1;
end;
writeln(', n, ' равен K = ',k);
End.
```



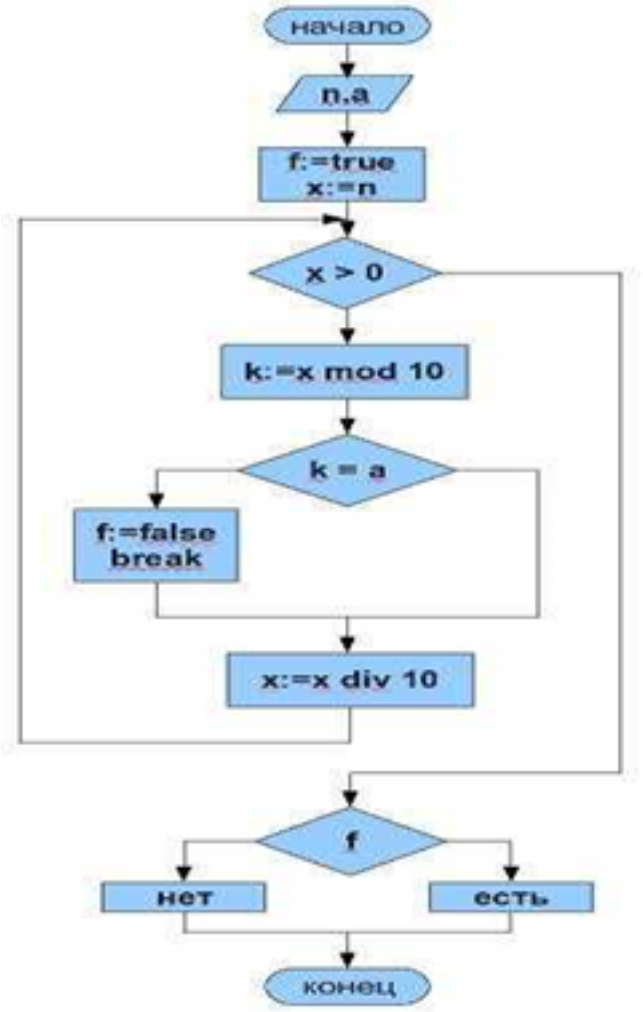
12. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



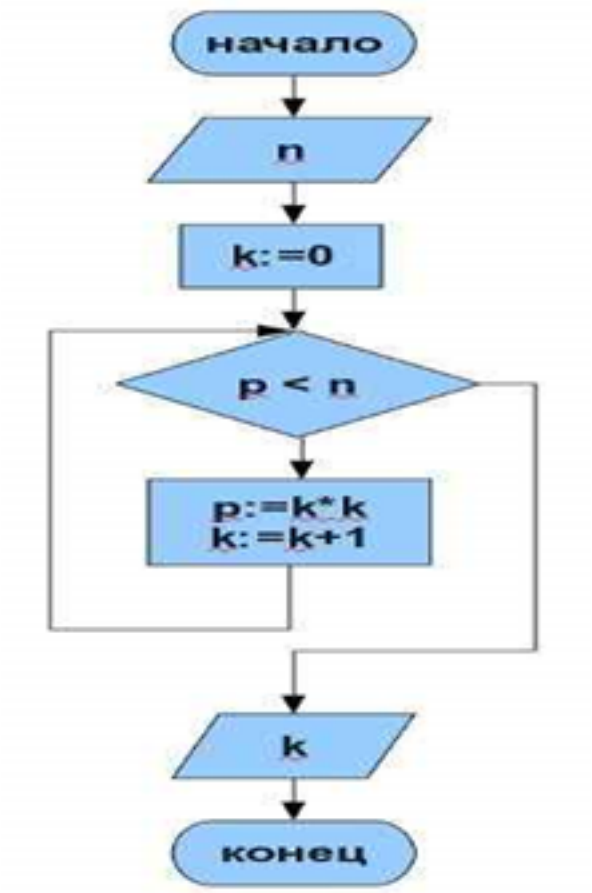
13. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



14. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



15. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



16. Выберите верное утверждение (1 балл)
    Переменная состоит из имени и выделенной области памяти, которая ему соответствует
    Числа являются частью переменных
    После присваивания переменная запоминает число
    Переменные в области памяти задаются независимо от средств вычисления и необходимостей автора. Невозможно изменить нахождения переменных в области памяти

17. Выберите верное утверждение (1 балл):
    При нахождением max и min перед циклом необходимо присвоить max ноль, min – максимальное число
    При нахождением max и min перед циклом необходимо присвоить max максимальное число, min – ноль
    При нахождением max и min перед циклом необходимо присвоить max, min – максимальное число
    При нахождением max и min перед циклом необходимо присвоить max, min – ноль

Оценка	*Показатели оценки*
3	Обучающийся выполнил 6 - 15 заданий
4	Обучающийся выполнил 16 - 19 заданий
5	Обучающийся выполнил 20 - 22 задания

Дидактическая единица: 2.1 разрабатывать алгоритмы для конкретных задач;
Занятие(-я):
2.1.5.Решение задач на составление разветвляющихся алгоритмов
2.2.3.Решение задач на составление циклических алгоритмов
2.2.4.Решение задач на составление циклических алгоритмов.
Задание №1
Разработать алгоритм работы программы в виде блок-схемы и программный код на любом языке программирования для следующей задачи:
Дано натуральное число. Определить произведение всех его цифр.

Оценка	*Показатели оценки*
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными.
4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Блок-схема соответствует выбранному алгоритму решения задачи.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует математическому решению произведения цифр числа. Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»; Блок-схема соответствует выбранному алгоритму решения задачи.

2.4 Текущий контроль (ТК) № 4

Тема занятия: 2.3.5.Составление алгоритмов на обработку одномерного массива. Сортировка элементов массива по заданному условию
Метод и форма контроля: Письменный опрос (Опрос)
Вид контроля: Проверочная работа
Дидактическая единица: 1.2 методы построения алгоритмов;
Занятие(-я):
2.3.1.Основные понятия об одномерном массиве. Поиск и замена элементов массива по заданному условию. Удаление и добавление элементов массива по заданному условию
2.3.2.Сортировка элементов одномерного массива по заданному условию
Задание №1
Выполните 5 теоретических заданий.
Каждое правильно выполненное задание оценивается в один балл. Возможен только один правильный ответ в каждом задании. Максимальное количество баллов - 5.

1. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Ниже представлен фрагмент программы, в котором значения элементов сначала задаются, а затем меняются.
   For i:=0 to 9 do
   A[i]:=9+I;
   For i:=0 to 4 do
   Begin
   K:=A[i]; A[i]:=A[9-i]; A[9-i]:=A[i]-k;
   End;
   Постройте блок-схему по программе. Чему будут равны элементы этого массива после выполнения фрагмента программы?
   1) 9 10 11 12 13 9 8 7 6 5
   2) 18 17 16 15 14 9 7 5 3 1
   3) 18 17 16 15 14 1 3 5 7 9
   4) 18 17 16 15 14 0 0 0 0 0

2. Определите, какое число будет напечатано в результате выполнения следующего алгоритма
   Var a, b, t, m, r: integer;
   Function F(x: integer): integer;
   begin
   F:=4(x+2)*(x-4);*
   End;

Begin
A:=-20; b:=20; m:=a; r:=F(a);
For t:=a to b d Begin
If (F(t)<r) then begin m:=t; r:=F(t); end;
End;
Write (m);
End.

3. Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:
   F(1) = 1,
   *F(n) = F(n-1)*n, при n>1.*
   Чему равно значение функции F(6)? В ответ запишите только натуральное число

4. Выберите верное утверждение:

Одномерный массив характеризуется множеством элементов, которые стоят в памяти, имеют одинаковый тип данных.

Одномерный массив характеризуется множеством элементов, которые стоят в памяти, имеют общее имя и одинаковый тип данных.

Одномерный массив характеризуется множеством элементов, которые стоят в памяти рядом, имеют общее имя и одинаковый тип данных.

5. Определите чему равен x после прохождения следующего алгоритма

```
a[5]={1,2,3,4,5};
for i:=1 to 5 do
a[i]=n-i+1;
a[1]:=a[2];
x:=a[1];
```

Оценка	***Показатели оценки***
3	Обучающийся выполнил 2-3 задания
4	Обучающийся выполнил 4 задания
5	Обучающийся выполнил 5 заданий

Дидактическая единица: 2.1 разрабатывать алгоритмы для конкретных задач;
Занятие(-я):
2.2.6.Решение задач на составление циклических алгоритмов. Вложенные циклы. Определение сложности циклических алгоритмов

2.3.3.Составление алгоритмов на обработку одномерного массива. Поиск и замена элементов массива по заданному условию.
2.3.4.Составление алгоритмов на обработку одномерного массива. Удаление и добавление элементов массива по заданному условию

Задание №1

Разработать алгоритм работы программы в виде блок-схемы и программный код на любом языке программирования для следующей задачи:
Дан одномерный массив из 20 чисел со значениями в диапазоне [-10;40].
Определить количество значений кратных 5.

Оценка	***Показатели оценки***
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Алгоритм решения задачи соответствует понятиям работы с одномерными массивами.
4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с одномерными массивами. Блок-схема соответствует выбранному алгоритму решения задачи.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с одномерными массивами. Алгоритм решения задачи соответствует математическому определению чисел кратных 5. Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»; Блок-схема соответствует выбранному алгоритму решения задачи.

2.5 Текущий контроль (ТК) № 5

Тема занятия: 2.4.7.Составление алгоритмов на обработку двумерного массива. Сортировка элементов массива по заданному условию
Метод и форма контроля: Контрольная работа (Опрос)
Вид контроля: Письменная контрольная работа
Дидактическая единица: 2.1 разрабатывать алгоритмы для конкретных задач;
Занятие(-я):
2.3.5.Составление алгоритмов на обработку одномерного массива. Сортировка элементов массива по заданному условию
2.4.3.Составление алгоритмов на обработку двумерного массива. Поиск и замена элементов массива по заданному условию
2.4.4.Составление алгоритмов на обработку двумерного массива. Удаление и добавление элементов массива по заданному условию
2.4.6.Составление алгоритмов на обработку двумерного массива. Сортировка элементов массива по заданному условию
Задание №1
Разработать программный код, который заполнит массив так, как показано на рисунке.

12	24	...	120
...	...	...	...
2	14	...	110
1	13	...	109

Оценка	***Показатели оценки***
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными.

4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами. Алгоритм решения задачи соответствует понятиям работы с памятью.

Задание №2

Разработать программный код для следующей задачи:

В зрительном зале 25 рядов. В каждом ряду 36 мест (кресел). Информация о проданных билетах хранится в двумерном массиве (номер строки – номер ряда, номер столбца – номер места). Если билет продан значение массива = 1, если нет, то значение= 0. Места распределите случайным образом и составьте программу, определяющую число проданных мест в 12-м ряду

После вывода ответа на вторую задачу поменяйте местами второй и предпоследний столбец местами.

Оценка	***Показатели оценки***
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными.

4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами. Алгоритм решения задачи соответствует понятиям работы с памятью.

2.6 Текущий контроль (ТК) № 6

Тема занятия: 2.5.4.Составление алгоритмов на обработку текстовых данных
Метод и форма контроля: Контрольная работа (Опрос)
Вид контроля: Письменная контрольная работа
Дидактическая единица: 1.3 методы вычисления сложности работы алгоритмов
Занятие(-я):
1.1.4.Оценка эффективности алгоритма.
2.2.5.Вложенные циклы. Вычисление сложности циклического алгоритма
Задание №1
Описать методы вычисления сложности работы алгоритмов.

Оценка	***Показатели оценки***

5	Обучающийся должен: I. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов. II. Описать каждый метод: 1) Вычислительная сложность - *подсчет количества выполняемых операций*. Точное количество операций будет зависеть от обрабатываемых данных, поэтому имеет смысл говорить о наилучшем, наихудшем и среднем случаях. (привести примеры расчета, достаточно - счетчик) 2) Сложность алгоритма по памяти: • память состоит из ячеек, каждая из которых имеет адрес и может хранить один элемент данных; • каждое обращение к памяти занимает одну единицу времени, независимо от номера адресуемой ячейки; • количество памяти достаточно для выполнения любого алгоритма; • процессор выполняет любую элементарную операцию за один временной шаг; • циклы и функции не считаются элементарными операциями.

4	Обучающийся должен I. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов. II. Описать один из двух методов: 1) Вычислительная сложность - *подсчет количества выполняемых операций*. Точное количество операций будет зависеть от обрабатываемых данных, поэтому имеет смысл говорить о наилучшем, наихудшем и среднем случаях. (привести примеры расчета, достаточно - счетчик) 2) Сложность алгоритма по памяти: • память состоит из ячеек, каждая из которых имеет адрес и может хранить один элемент данных; • каждое обращение к памяти занимает одну единицу времени, независимо от номера адресуемой ячейки; • количество памяти достаточно для выполнения любого алгоритма; • процессор выполняет любую элементарную операцию за один временной шаг; • циклы и функции не считаются элементарными операциями.
3	Обучающийся должен указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов.

Дидактическая единица: 2.2 определять сложность работы алгоритмов;
Занятие(-я):
2.2.6.Решение задач на составление циклических алгоритмов. Вложенные циклы. Определение сложности циклических алгоритмов

Задание №1

Дан программный код, определить сложность работы алгоритма:

```
min := array[1]
```

```
for i:= 2 to N do
  if array[i] < min then min := array[i];
write(min);
```

Оценка	***Показатели оценки***
3	Рассчитана сложность работы алгоритма одним из методов (*вычислительная сложность* или *сложность алгоритма по памяти)* с использованием таблицы определения сложности алгоритма.
4	Рассчитана сложность работы алгоритма одним из методов (*вычислительная сложность* или *сложность алгоритма по памяти)* без использования таблицы определения сложности алгоритма.
5	Рассчитана сложность работы алгоритма двумя методами (*вычислительная сложность* или *сложность алгоритма по памяти)* без использования таблицы определения сложности алгоритма.

3. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

№ семестра	Вид промежуточной аттестации
3	Экзамен

Экзамен может быть выставлен автоматически по результатам текущих контролей
Текущий контроль №1
Текущий контроль №2
Текущий контроль №3
Текущий контроль №4
Текущий контроль №5
Текущий контроль №6

Метод и форма контроля: Контрольная работа (Опрос)
Вид контроля: экзаменационный билет содержит одно теоретическое и два практических задания
Дидактическая единица для контроля:
1.1 основные модели алгоритмов;
Задание №1 (из текущего контроля)
Решите тест из 20 вопросов.
Каждый правильный ответ на вопрос оценивается в один балл. Возможен только один правильный ответ в каждом вопросе. Максимальное количество баллов - 20.

1. Что такое алгоритм?
   набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий.
   инсталляция порядка действий для ЭВМ, который составлен для определенного языка программирования и удовлетворяет определенным правилам.
   инсталляция порядка действий для ЭВМ, который составлен и удовлетворяет определенным правилам
   искусство счета с помощью цифр, но поначалу слово «цифра» относилось только к нулю

2. Какое из перечисленных свойств не относится к свойствам алгоритмов?
   Дискретность Определенность Результативность Полнота Массовость

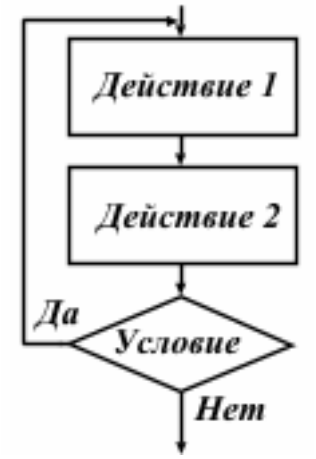
Правильность

3. Какое свойство алгоритмов характеризует данное определение: «Алгоритм должен приводить к решению за конечное число шагов»
   Дискретность Определенность Результативность Полнота Массовость Правильность

4. Какое свойство алгоритмов характеризует данное определение: «Каждое правило алгоритма должно быть четким, однозначным»
   Дискретность Определенность Результативность Полнота Массовость Правильность

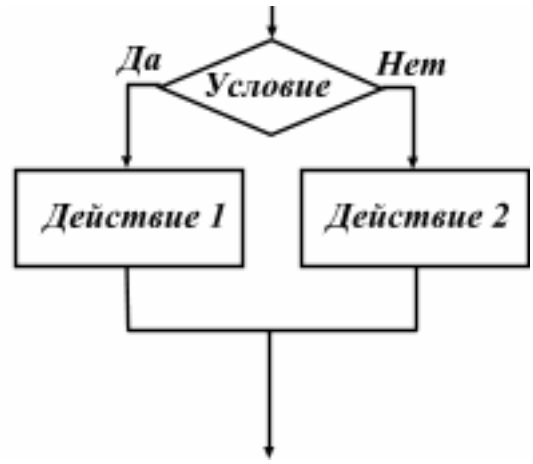
5. Какое свойство алгоритмов характеризует данное определение: «Алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными»
   Дискретность Определенность Результативность Полнота Массовость Правильность

6. Какой вид алгоритма представлен на рисунке?



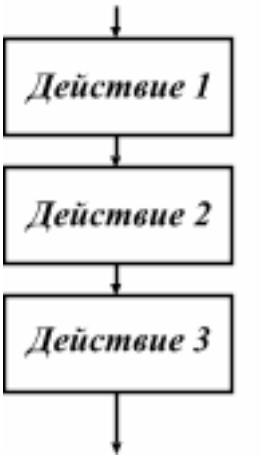
   Линейный Разветвляющий Циклический

7. Какой вид алгоритма представлен на рисунке?



Линейный Разветвляющий Циклический

8. Какой вид алгоритма представлен на рисунке?



Линейный Разветвляющий Циклический

9. Выберите вариант, где представлены все способы описания алгоритма (но нет лишних).
   Словесный, Формульно-словесный, Графический, Программный
   Словесный, Формульный, Формульно-словесный, Графический, Программный
   Словесный, Формульный, Графический, Программный
   Формульный, Графический, Программный

10. Что такое блок схема?
    распространенный тип схем, описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.
    схематичное изображение элементов алгоритма, которые собранны воедино с помощью линий в определенном порядке, демонстрируя алгоритм.
    строгое изображение элементов алгоритма, которые собранны воедино по

правилам с помощью линий в определенном порядке, демонстрируя алгоритм.
схема программы на языке программирования, представленная в виде блоков, соединенных линиями

11. Блок схема. На рисункепредставлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

12. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

13. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

14. Блок схема. На рисунке представлено обозначение элемента алгоритма. Выберите его наименование



Блок начало-конец
Блок действия
Логический блок (блок условия)
Данные

15. В чем заключается Разработка алгоритма решения задачи
в установлении необходимой последовательности арифметических и логических действий, строгое выполнение которых приводит к решению задачи.
строгое выполнение арифметических и логических задач в необходимой последовательности в программном коде
выполнение арифметических и логических задач в необходимой последовательности в программном коде несет рекомендательный характер

16. Вставьте пропущенное слово: « Любой алгоритм применяется к … и выдает результат»
исходным данным
внешним данным
используемым средствам
внутренним средствам

17. Вставьте пропущенное слово: «Данные для своего размещения требуют …. … обычно считается дискретной. Единицы измерения … и данных должны быть согласованы между собой»
память
элементарный шаг
участок
промежуток

18. Вставьте пропущенное слово: «Алгоритм состоит из отдельных …»
Участков памяти
элементарный шагов
участков
промежутков

19. Вставьте пропущенное слово: «Последовательность шагов алгоритма …, т.е. после каждого шага указывается следующий шаг, либо алгоритм

останавливается»
детерминирована
результативна
объективна
полна

20. Вставьте пропущенное слово: «Каждый алгоритм должен быть … , т.е. после конечного числа шагов выдавать результат»
детерминировным
результативным
объективным
полным

Оценка	***Показатели оценки***
3	Обучающийся ответил на 6-13 вопросов
4	Обучающийся ответил на 14-17 вопросов
5	Обучающийся ответил на 18-20 вопросов

Дидактическая единица для контроля:
1.2 методы построения алгоритмов;
Задание №1 (из текущего контроля)
Выполните 17 теоретических заданий (в трех первых заданиях ответ - действительные числа, в остальных 14 - выбор ответа).
Каждое правильно выполненное задание оценивается от 1 до 3 баллов (количество баллов указано в формулировке каждого задания). Возможен только один правильный ответ в каждом задании. Максимальное количество баллов - 22.

1. Определите, что будет напечатано в результате выполнения программы (2 балла)

```
var n, s: integer;
begin
  n := 0;
  s := 512;
  while s >= 0 do
  begin
    s := s - 20;
    n := n + 1
  end;
  write(n)
end.
```

2. Получив на вход число x, этот алгоритм печатает два числа *a* и *b*. Укажите НАИБОЛЬШЕЕ из таких чисел x, при вводе которых алгоритм печатает сначала 2, а потом 8 (3 балла)

```
var x, a, b: integer;
begin
  readln(x);
  a:=0; b:=0;
  while x>0 do
  begin
    a:=a+1;
    b:=b+(x mod 10);
    x:=x div 10;
  end;
  writeln(a); write(b);
end.
```

3. Определите, что будет напечатано в результате выполнения программы (2 балла)

```
var n, s: integer;

begin

  n := 24;

  s := 0;

  while n <= 28 do

  begin

    s := s + 20;

    n := n + 2

  end;

  write(s)

end.
```

4. Задача: даны две цифры, необходимо узнать какая из них больше и вывести ответ на экран. Какого вида алгоритма будет достаточно для решения данной задачи? (1 балл)
   Линейного
   Разветвляющего
   Циклического

5. Что такое Цикл (1 балл)
   - это последовательность операторов, которая может выполняться один или более раз
   - это последовательность переменных, которые выполняются один или более раз
   - это последовательность переменных, которые выполняются хаотичное число раз
   - это последовательность операторов, которая может выполняться хаотичное раз

6. К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (while)» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)

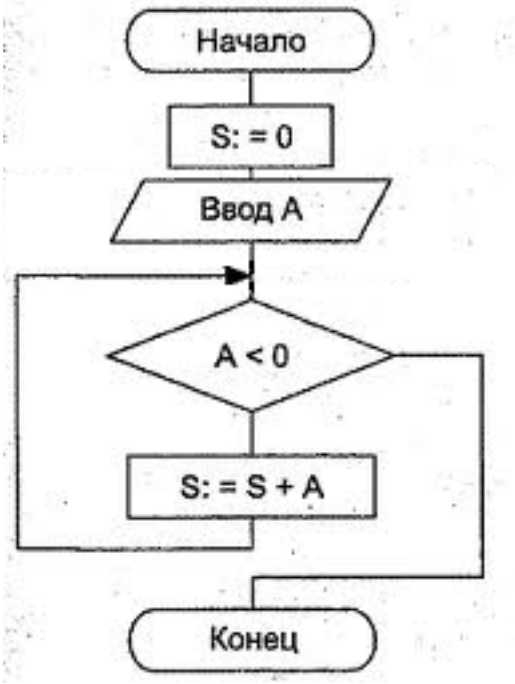
Вложенные циклы

7. К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (repeat)» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

8. К какому виду цикла относится данное определение: «используется тогда, когда заранее определенное число повторений» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

9. К какому виду цикла относится данное определение: «находится в теле другого цикла» (1 балл)
   Цикл с предусловием
   Цикл с постусловием
   Цикл с параметром (счетчик)
   Вложенные циклы

10. На Картинке представлена блок-схема. Выберите код данной блок-схемы. (1 балл)



```
Program;
var S;A:=integer;
begin
S:=0;
readln(A);
while A<0 do
begin
S:=S+A;
end;
end.
```

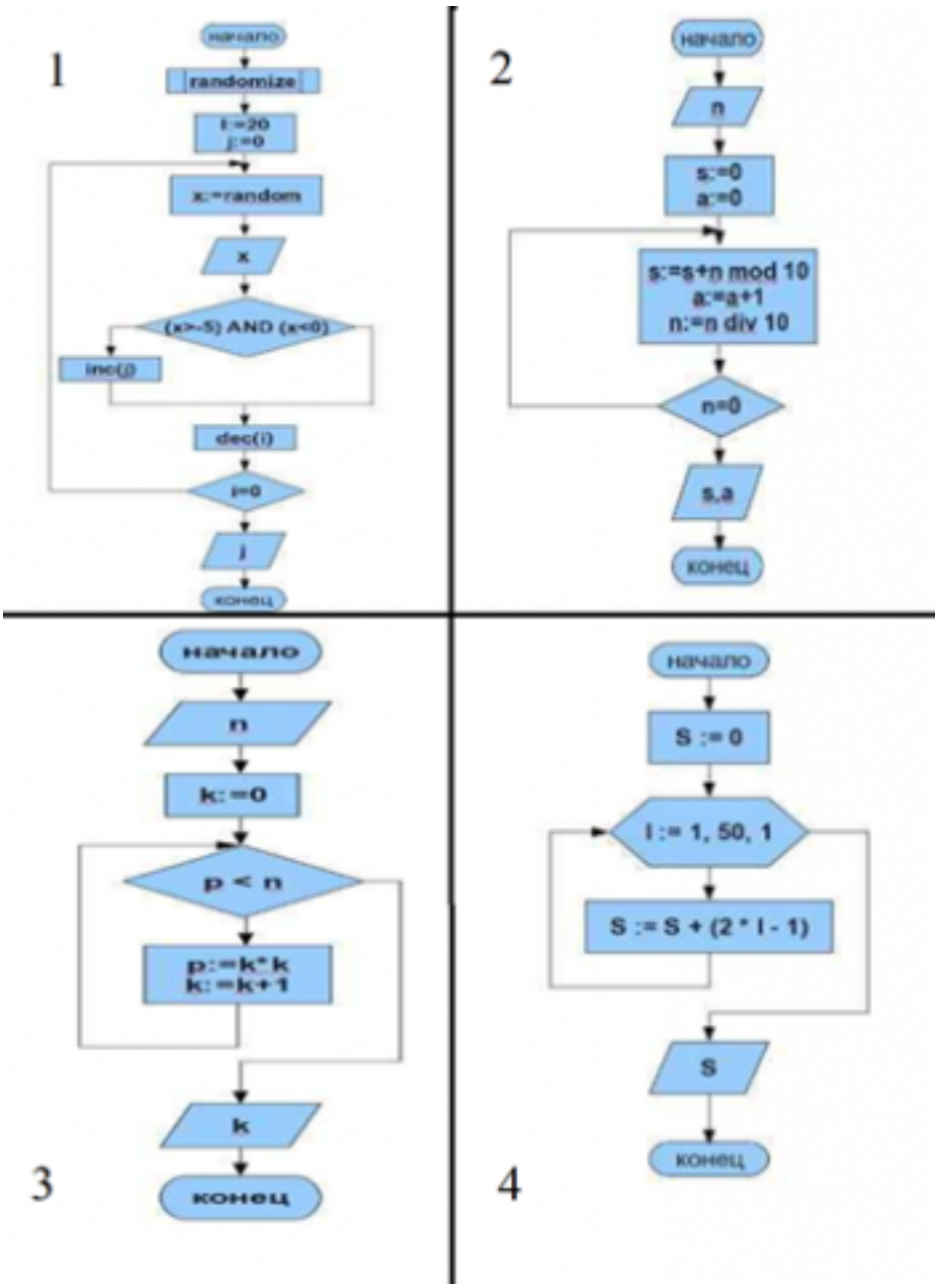
```
Program;
var S;A:=integer;
begin
readln(A);
S:=0;
while A<0 do
begin
S:=S+A;
end;
end.
```

```
Program;
var S;A:=integer;
begin
S:=0;
readln(A);
repeat
S:=S+A; until A<0
end;
end.
```

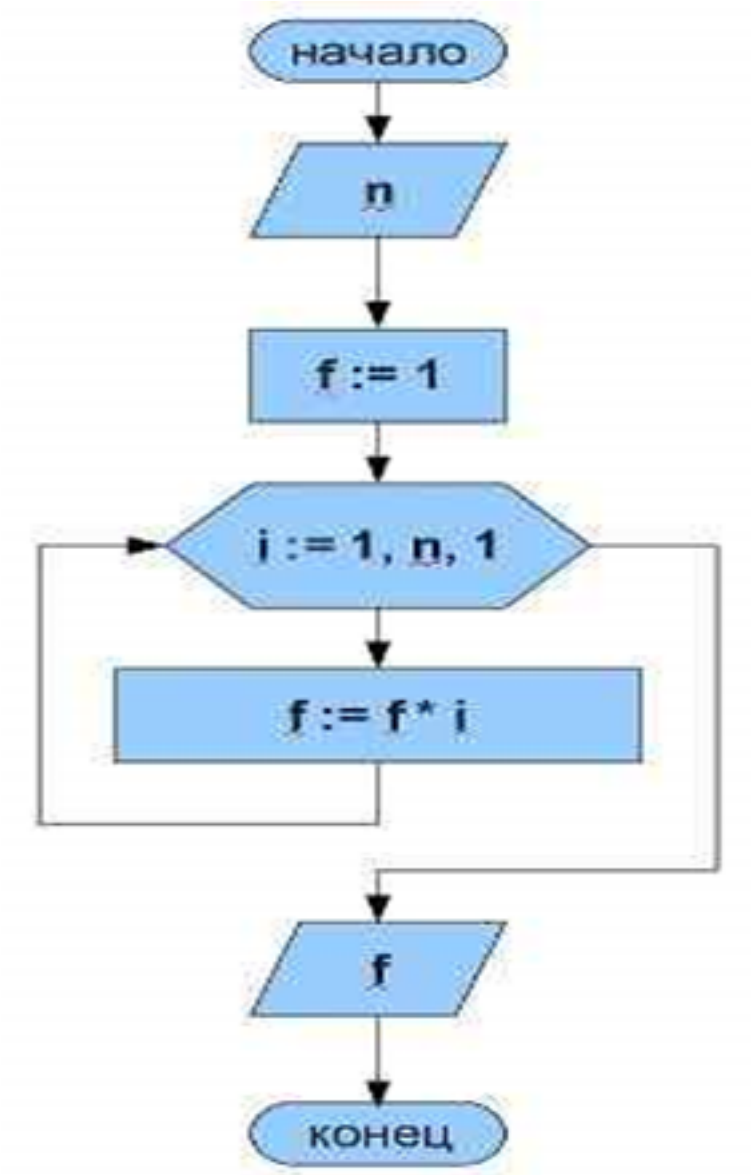
```
Program;
var S;A:=integer;
begin
readln(A);
S:=0;
repeat
S:=S+A; until A<0
end;
end.
```

11. На картинке представлены блок-схемы, выберете нужную по данной программе: (2 балла)

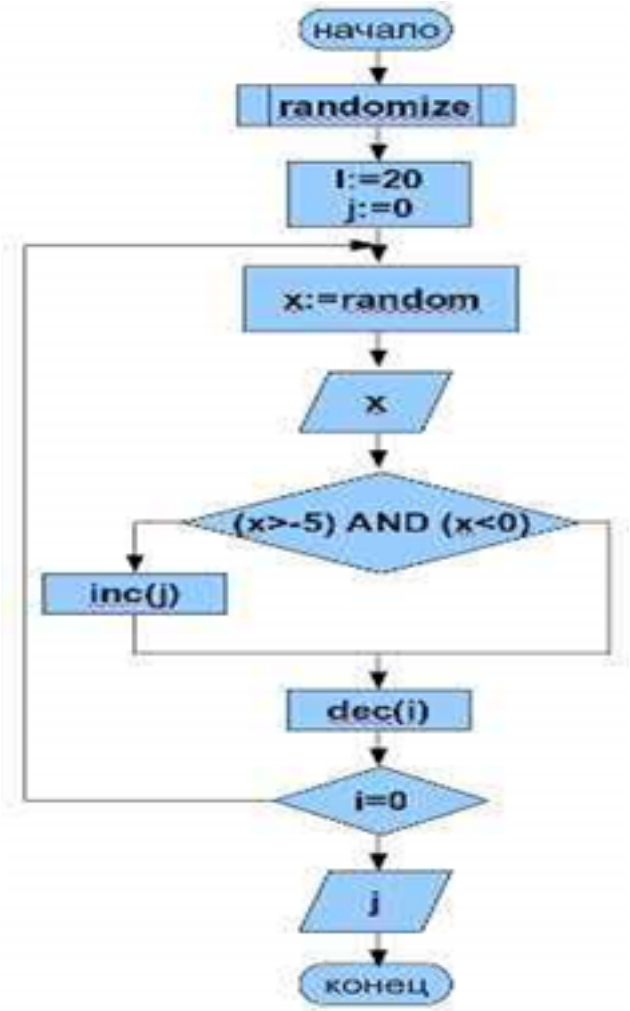
```
program ;
Var
  n,k,p:longint;
Begin
Write(‘Введите число N > 0, N = '); readln(n);
k:=0;
while p < n Do
Begin
  p := k * k;
  k := k + 1;
end;
writeln(', n, ' равен K = ',k);
End.
```



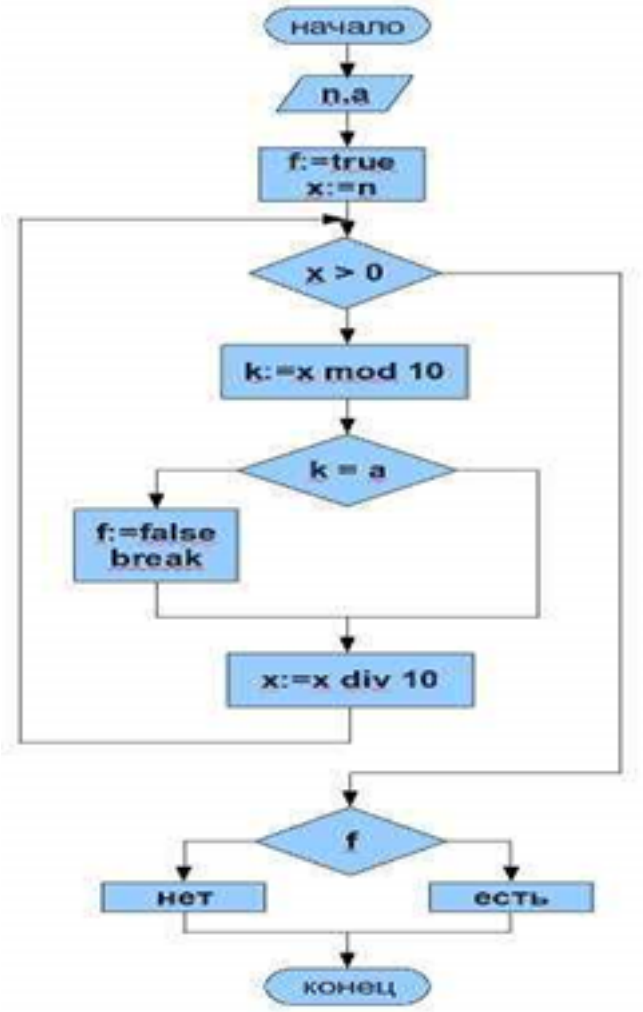
12. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



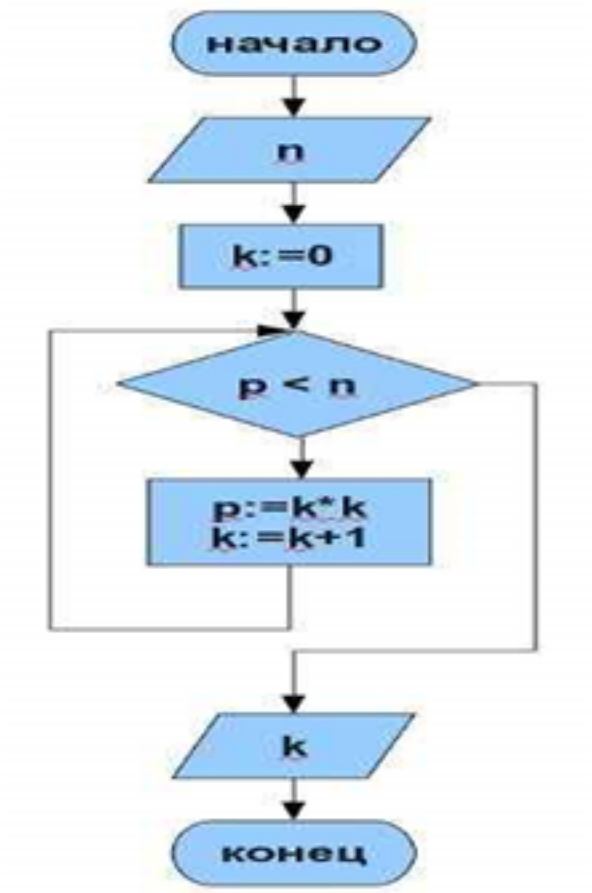
13. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



14. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



15. На картинке представлена блок-схема. Выберите нужный оператор для построения кода. (1 балл)
    " while"
    " repeat"
    " for"
    " joy"



16. Выберите верное утверждение (1 балл)
    Переменная состоит из имени и выделенной области памяти, которая ему соответствует
    Числа являются частью переменных
    После присваивания переменная запоминает число
    Переменные в области памяти задаются независимо от средств вычисления и необходимостей автора. Невозможно изменить нахождения переменных в области памяти

17. Выберите верное утверждение (1 балл):
    При нахождением max и min перед циклом необходимо присвоить max ноль, min – максимальное число
    При нахождением max и min перед циклом необходимо присвоить max максимальное число, min – ноль
    При нахождением max и min перед циклом необходимо присвоить max, min – максимальное число
    При нахождением max и min перед циклом необходимо присвоить max, min – ноль

Оценка	*Показатели оценки*
3	Обучающийся выполнил 6 - 15 заданий
4	Обучающийся выполнил 16 - 19 заданий
5	Обучающийся выполнил 20 - 22 задания

Дидактическая единица для контроля:

1.3 методы вычисления сложности работы алгоритмов

Задание №1 (из текущего контроля)

Описать методы вычисления сложности работы алгоритмов.

Оценка	*Показатели оценки*
5	Обучающийся должен: I. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов. II. Описать каждый метод: 1) Вычислительная сложность - *подсчет количества выполняемых операций*. Точное количество операций будет зависеть от обрабатываемых данных, поэтому имеет смысл говорить о наилучшем, наихудшем и среднем случаях. (привести примеры расчета, достаточно - счетчик) 2) Сложность алгоритма по памяти: • память состоит из ячеек, каждая из которых имеет адрес и может хранить один элемент данных; • каждое обращение к памяти занимает одну единицу времени, независимо от номера адресуемой ячейки; • количество памяти достаточно для выполнения любого алгоритма; • процессор выполняет любую элементарную операцию за один временной шаг; • циклы и функции не считаются элементарными операциями.

4	Обучающийся должен I. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов. II. Описать один из двух методов: 1) Вычислительная сложность - *подсчет количества выполняемых операций*. Точное количество операций будет зависеть от обрабатываемых данных, поэтому имеет смысл говорить о наилучшем, наихудшем и среднем случаях. (привести примеры расчета, достаточно - счетчик) 2) Сложность алгоритма по памяти: • память состоит из ячеек, каждая из которых имеет адрес и может хранить один элемент данных; • каждое обращение к памяти занимает одну единицу времени, независимо от номера адресуемой ячейки; • количество памяти достаточно для выполнения любого алгоритма; • процессор выполняет любую элементарную операцию за один временной шаг; • циклы и функции не считаются элементарными операциями.
3	Обучающийся должен указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов.

Дидактическая единица для контроля:

2.1 разрабатывать алгоритмы для конкретных задач;

Задание №1 (из текущего контроля)

Разработать программный код для следующей задачи:

В зрительном зале 25 рядов. В каждом ряду 36 мест (кресел). Информация о проданных билетах хранится в двумерном массиве (номер строки – номер ряда, номер столбца – номер места). Если билет продан значение массива = 1, если нет, то

значение= 0. Места распределите случайным образом и составьте программу, определяющую число проданных мест в 12-м ряду
После вывода ответа на вторую задачу поменяйте местами второй и предпоследний столбец местами.

Оценка	***Показатели оценки***
3	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными.
4	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами.
5	Необходимо выполнить следующие операции и приемы: Описание входных данных (их типов, диапазонов); Описание выходных данных; Запись математического соотношения, связывающего результат с исходными данными. Алгоритм решения задачи соответствует понятиям работы с двумерными массивами. Алгоритм решения задачи соответствует понятиям работы с памятью.

Дидактическая единица для контроля:
2.2 определять сложность работы алгоритмов;
Задание №1 (из текущего контроля)
Дан программный код, определить сложность работы алгоритма:

```
min := array[1]
for i:= 2 to N do
  if array[i] < min then min := array[i];
write(min);
```

Оценка	***Показатели оценки***

3	Рассчитана сложность работы алгоритма одним из методов (*вычислительная сложность* или *сложность алгоритма по памяти)* с использованием таблицы определения сложности алгоритма.
4	Рассчитана сложность работы алгоритма одним из методов (*вычислительная сложность* или *сложность алгоритма по памяти)* без использования таблицы определения сложности алгоритма.
5	Рассчитана сложность работы алгоритма двумя методами (*вычислительная сложность* или *сложность алгоритма по памяти)* без использования таблицы определения сложности алгоритма.