



Областное государственное образовательное  
учреждение среднего профессионального  
образования «Иркутский авиационный  
техникум»

«УТВЕРЖДАЮ»  
Директор  
ОГБОУ СПО "ИАТ"  
В.Г. Семенов В.Г. Семенов  
«31» августа 2013 г.

**Фонд оценочных средств  
по дисциплине**

ОП.08 Теория алгоритмов

образовательной программы  
по специальности СПО

09.02.03 Программирование в компьютерных системах  
базовой подготовки

г.Иркутск

Рассмотрена  
цик洛вой комиссией

Протокол № \_\_\_\_\_  
от «\_\_\_\_» \_\_\_\_ 20\_\_ г.

Председатель ЦК  
\_\_\_\_\_ / \_\_\_\_\_ /

№	Разработчик ФИО (полностью)
1	Пашкевич Виктория Владиславовна

# **1. ОБЩИЕ ПОЛОЖЕНИЯ**

## **1.1. Область применения фонда оценочных средств (ФОС)**

ФОС по дисциплине является частью образовательной программы в соответствии с ФГОС СПО по специальности  
09.02.03 Программирование в компьютерных системах

## **1.2. Место дисциплины в структуре ОПОП:**

ОП.00 Общепрофессиональные дисциплины

## **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.

ОК.10 Исполнять воинскую обязанность, в том числе с применением полученных профессиональных знаний (для юношей).

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: (2.2.1. Циклические алгоритмы. Цикл с параметром)**

Дидактические единицы	Основные показатели оценивания результата	№ задания относящийся к показателю оценивания	Метод контроля	Форма контроля	Вид контроля	Индексы занятий ранее изученных связанные с контролируемыми дидактическими единицами
1.1	1.1.1, 1.1.2, 1.1.3	1.1, 1.2, 1.3, 1.4, 1.5, 1.9, 1.10, 1.15, 1.16, 1.17, 1.18, 1.19, 1.20, 1.6, 1.7, 1.8, 1.11, 1.12, 1.13, 1.14	Опрос	Тестирование	Письменное тестирование	1.1.1, 1.1.2

### **Перечень заданий текущего контроля**

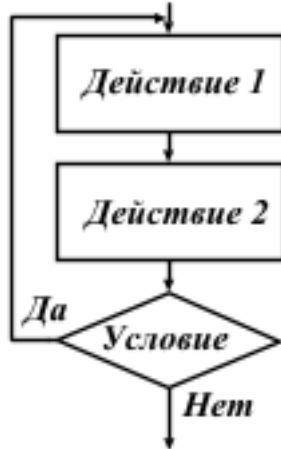
Номер задания	Задания
1.1	Что такое алгоритм?  набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи

	<p>за конечное число действий.</p> <p>инсталляция порядка действий для ЭВМ, который составлен для определенного языка программирования и удовлетворяет определенным правилам.</p> <p>инсталляция порядка действий для ЭВМ, который составлен и удовлетворяет определенным правилам</p> <p>искусство счёта с помощью цифр, но поначалу слово «цифра» относилось только к нулю</p>
1.2	<p>Какое из перечисленных свойств не относится к свойствам алгоритмов?</p> <p>Дискретность Определенность Результативность Полнота Массовость Правильность</p>
1.3	<p>Какое свойство алгоритмов характеризует данное определение: «Алгоритм должен приводить к решению за конечное число шагов»</p> <p>Дискретность Определенность Результативность Полнота Массовость Правильность</p>
1.4	<p>Какое свойство алгоритмов характеризует данное определение: «Каждое правило алгоритма должно быть четким, однозначным»</p> <p>Дискретность Определенность Результативность Полнота Массовость Правильность</p>
1.5	<p>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»</p> <p>Дискретность Определенность Результативность Полнота Массовость Правильность</p>
1.9	Выберите вариант, где представлены все способы описания алгоритма (но нет лишних).

	<p>Словесный, Формульно-словесный, Графический, Программный</p> <p>Словесный, Формульный, Формульно-словесный, Графический, Программный</p> <p>Словесный, Формульный, Графический, Программный</p> <p>Формульный, Графический, Программный</p>
1.10	<p><b>Что такое блок схема?</b></p> <p>распространенный тип схем, описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.</p> <p>схематичное изображение элементов алгоритма, которые собраны воедино с помощью линий в определенном порядке, демонстрируя алгоритм.</p> <p>строгое изображение элементов алгоритма, которые собраны воедино по правилам с помощью линий в определенном порядке, демонстрируя алгоритм.</p> <p>схема программы на языке программирования, представленная в виде блоков, соединенных линиями</p>
1.15	<p>В чем заключается <b>Разработка алгоритма решения задачи</b></p> <p>в установлении необходимой последовательности арифметических и логических действий, строгое выполнение которых приводит к решению задачи.</p> <p>строгое выполнение арифметических и логических задач в необходимой последовательности в программном коде</p>

	выполнение арифметических и логических задач в необходимой последовательности в программном коде несет рекомендательный характер
1.16	<p>Вставьте пропущенное слово: «Любой алгоритм применяется к ... и выдает результат»</p> <p><b>исходным данным</b></p> <p>внешним данным</p> <p>используемым средствам</p> <p>внутренним средствам</p>
1.17	<p>Вставьте пропущенное слово: «Данные для своего размещения требуют .... . . . обычно считается дискретной. Единицы измерения ... и данных должны быть согласованы между собой»</p> <p><b>память</b></p> <p>элементарный шаг</p> <p>участок</p> <p>промежуток</p>
1.18	<p>Вставьте пропущенное слово: «Алгоритм состоит из отдельных ...»</p> <p><b>Участков памяти</b></p> <p>элементарный шагов</p>

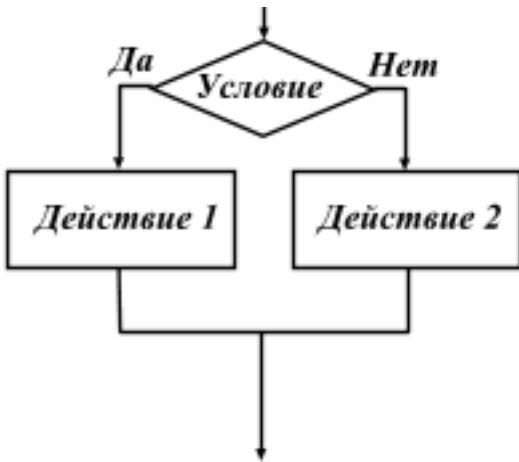
	участков промежутков
1.19	<p>Вставьте пропущенное слово: «Последовательность шагов алгоритма ..., т.е. после каждого шага указывается следующий шаг, либо алгоритм останавливается»</p> <p><b>детерминирована</b></p> <p><b>результативна</b></p> <p><b>объективна</b></p> <p><b>полна</b></p>
1.20	<p>Вставьте пропущенное слово: «Каждый алгоритм должен быть ..., т.е. после конечного числа шагов выдавать результат»</p> <p><b>детерминированным</b></p> <p><b>результативным</b></p> <p><b>объективным</b></p> <p><b>полным</b></p>
1.6	Какой вид алгоритма представлен на рисунке (нажмите клавишу «Рисунок»)?



Линейный Разветвляющий Циклический

1.7

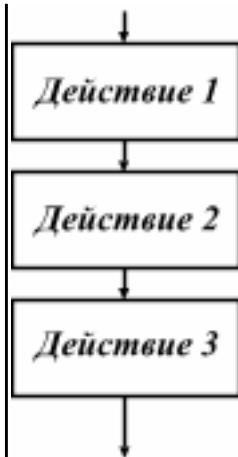
Какой вид алгоритма представлен на рисунке (нажмите клавишу «Рисунок»)?



Линейный Разветвляющий Циклический

1.8

Какой вид алгоритма представлен на рисунке (нажмите клавишу «Рисунок»)?



Линейный Разветвляющий Циклический

- 1.11 Блок схема. На рисунке (нажмите клавишу) представлено обозначение элемента алгоритма. Выберите его наименование



**Блок начало-конец**

**Блок действия**

**Логический блок (блок условия)**

**Данные**

- 1.12 Блок схема. На рисунке (нажмите клавишу) представлено обозначение элемента алгоритма. Выберите его

	<p>наименование</p> 
	<p><b>Блок начало-конец</b></p>
	<p><b>Блок действия</b></p>
	<p><b>Логический блок (блок условия)</b></p>
	<p><b>Данные</b></p>
1.13	<p>Блок схема. На рисунке (нажмите клавишу) представлено обозначение элемента алгоритма. Выберите его наименование</p> 
	<p><b>Блок начало-конец</b></p>
	<p><b>Блок действия</b></p>
	<p><b>Логический блок (блок условия)</b></p>

	<b>Данные</b>
1.14	<p>Блок схема. На рисунке (нажмите клавишу) представлено обозначение элемента алгоритма. Выберите его наименование</p>  <p><b>Блок начало-конец</b></p> <p><b>Блок действия</b></p> <p><b>Логический блок (блок условия)</b></p> <p><b>Данные</b></p>

#### Перечень показателей текущего контроля

Номер показателя	Значение показателя
1.1.1	Выявление правильного определения основных моделей алгоритмов
1.1.2	Определение вида основной модели алгоритмов по рисунку
1.1.3	Определение наименования элемента блок-схемы основных моделей алгоритмов

## 2.2 Текущий контроль (ТК) № 2

**Тема занятия: (2.4.1.Основные понятия о двумерном массиве. Поиск и замена элементов массива по заданному условию. Удаление и добавление элементов массива по заданному условию)**

Дидактические единицы	Основные показатели оценивания результата	№ задания относящийся к показателю оценивания	Метод контроля	Форма контроля	Вид контроля	Индексы занятий ранее изученных связанные с контролируемыми единицами
1.2	1.2.1, 1.2.2, 1.2.3, 1.2.4	2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.12, 2.13, 2.14, 2.15, 2.16, 2.17, 2.20, 2.7, 2.8, 2.18, 2.19, 2.9, 2.10, 2.11	Опрос	Тестирование	Письменное тестирование	1.1.2, 1.1.3, 2.1.1, 2.1.3, 2.2.1, 2.2.2, 2.2.5, 2.3.1, 2.3.2

### Перечень заданий текущего контроля

Номер задания	Задания
2.1	Определите значение переменной с 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;
```

- 2.2 Определите, что будет напечатано в результате выполнения программы

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

```

- 2.3 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

```
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.
```

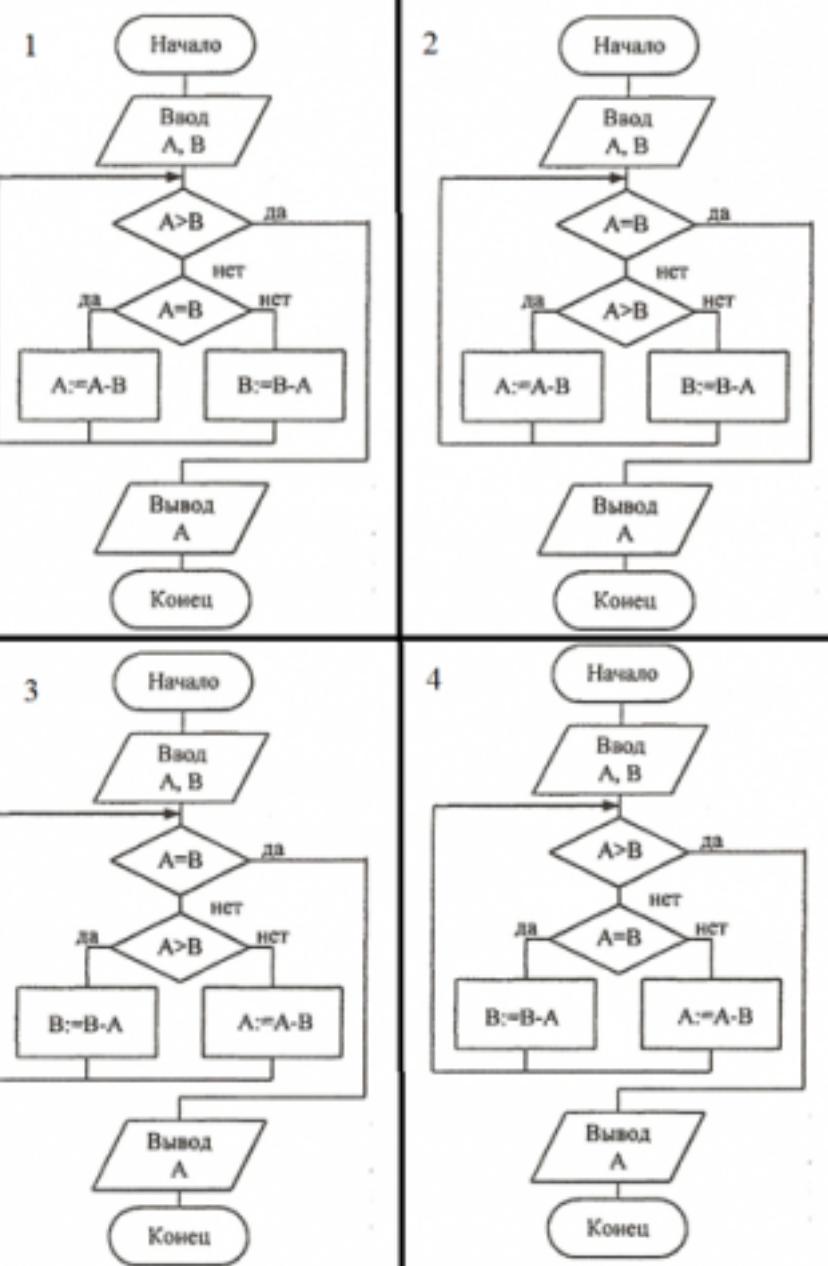
- 2.4 Определите значение переменной с после выполнения следующего фрагмента программы

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

- 2.5 Определите, что будет напечатано в результате выполнения программы

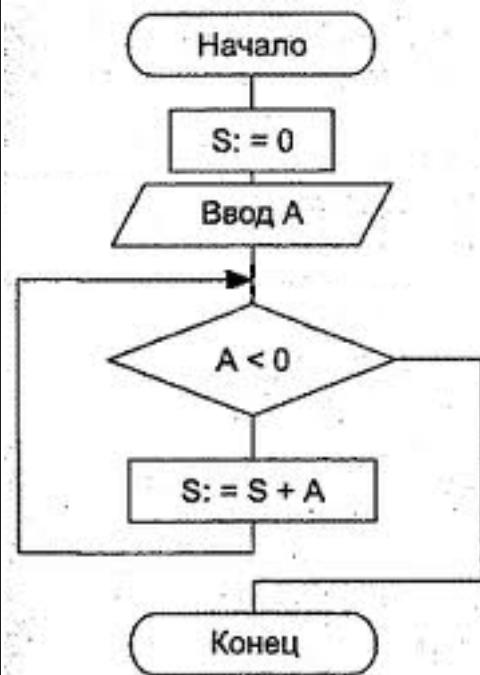
```
var n, s: integer;  
  
begin  
    n := 24;  
    s := 0;  
    while n <= 28 do  
    begin  
        s := s + 20;  
        n := n + 2  
    end;  
    write(s)  
end.
```

- 2.6 На рисунке представлены алгоритмы. Только один из них алгоритм Евклида. Какой?



2.12

На Картинке представлена блок-схема. Выберите код данной блок-схемы.



```
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.
```

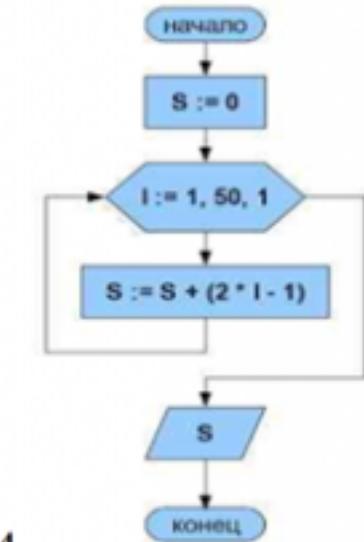
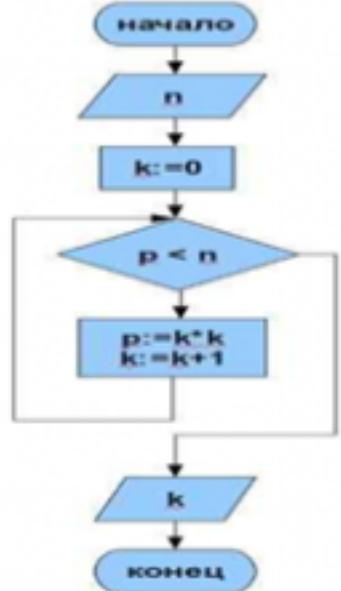
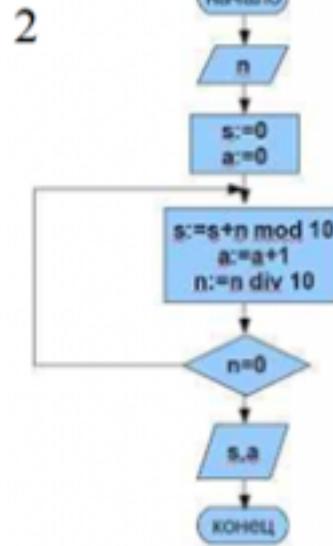
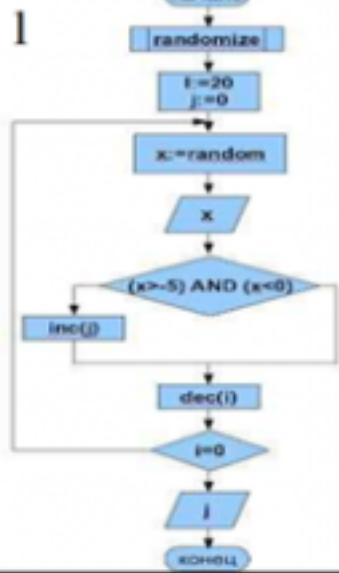
```
Program;
var S;A:=integer;
begin
```

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

2.13 На картинке представлены блок-схемы, выберете нужную по данной программе:

```
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.
```



2.14

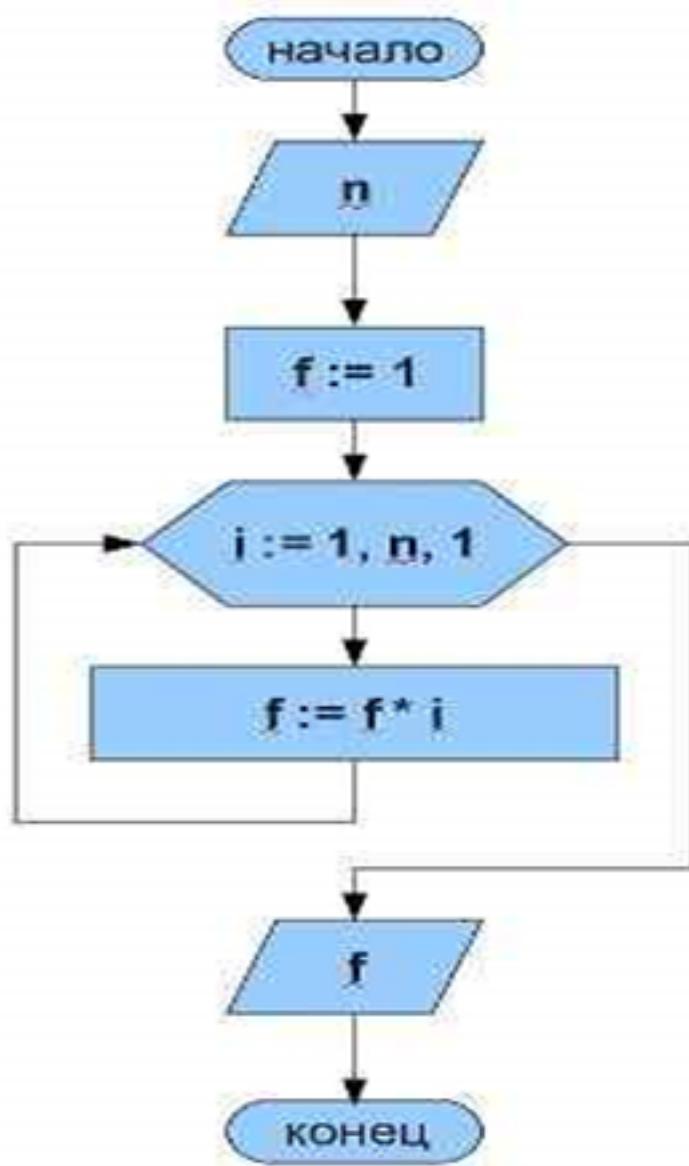
На картинке представлена блок-схема. Выберите нужный оператор для построения кода.

" while"

" repeat"

" for"

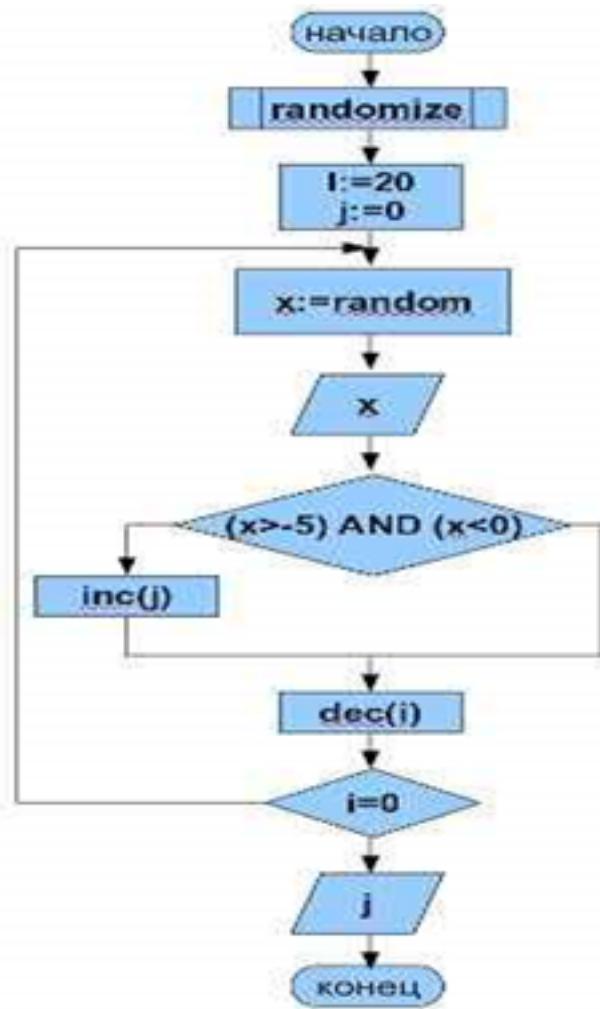
" joy"



2.15

На картинке представлена блок-схема. Выберите нужный оператор для построения кода.

```
" while"  
" repeat"  
" for"  
" joy"
```

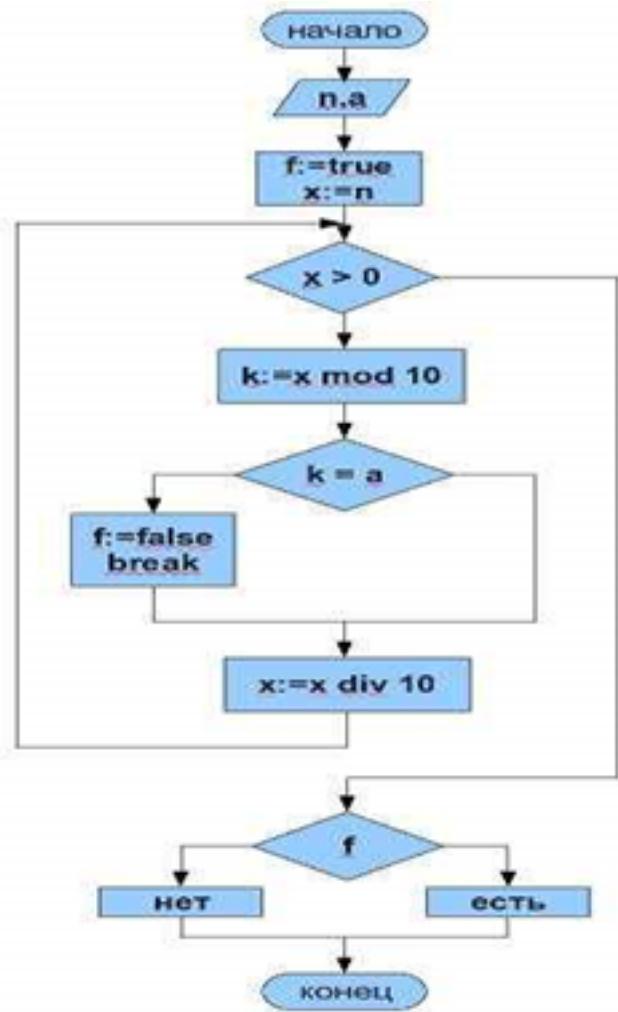


2.16 На картинке представлена блок-схема. Выберите нужный оператор для построения кода.

" while"

" repeat"

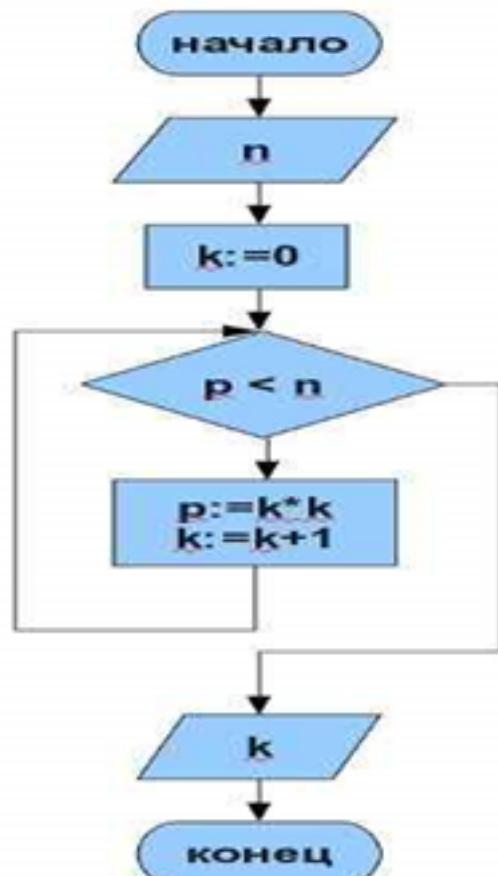
```
" for"
" joy"
```



2.17

На картинке представлена блок-схема. Выберите нужный оператор для построения кода.

```
" while"  
" repeat"  
" for"  
" joy"
```



2.20	<p>Задача: даны две цифры, необходимо узнать какая из них больше и вывести ответ на экран. Какого вида алгоритма будет достаточно для решения данной задачи?</p> <p>Линейного</p> <p>Разветвляющего</p> <p>Циклического</p>
2.7	<p>Что такое Цикл</p> <ul style="list-style-type: none"> <li>- это последовательность операторов, которая может выполняться один или более раз</li> <li>- это последовательность переменных, которые выполняются один или более раз</li> <li>- это последовательность переменных, которые выполняются хаотичное число раз</li> <li>- это последовательность операторов, которая может выполнятся хаотичное раз</li> </ul>
2.8	<p>К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (while)»</p> <p>Цикл с предусловием</p> <p>Цикл с постусловием</p> <p>Цикл с параметром (счетчик)</p>

## Вложенные циклы

2.18	<p>Выберите верное утверждение</p> <p><b>Переменная</b> состоит из имени и выделенной области памяти, которая ему соответствует</p> <p><b>Числа являются частью переменных</b></p> <p>После присваивания переменная запоминает число</p> <p>Переменные в области памяти задаются независимо от средств вычисления и необходимостей автора. Невозможно изменить нахождения переменных в области памяти</p>
2.19	<p>Выберите верное утверждение:</p> <p>При нахождением <code>max</code> и <code>min</code> перед циклом необходимо присвоить <code>max</code> ноль, <code>min</code> – максимальное число</p> <p>При нахождением <code>max</code> и <code>min</code> перед циклом необходимо присвоить <code>max</code> максимальное число, <code>min</code> – ноль</p> <p>При нахождением <code>max</code> и <code>min</code> перед циклом необходимо присвоить <code>max</code>, <code>min</code> – максимальное число</p> <p>При нахождением <code>max</code> и <code>min</code> перед циклом необходимо присвоить <code>max</code>, <code>min</code> – ноль</p>
2.9	<p>К какому виду цикла относится данное определение: «используются тогда, когда не известно, сколько раз нужно выполнить тело цикла (<code>repeat</code>)»</p>
2.10	<p>К какому виду цикла относится данное определение: «используется тогда, когда заранее определенное число повторений»</p>
2.11	<p>К какому виду цикла относится данное определение: «находится в теле другого цикла»</p>

### Перечень показателей текущего контроля

Номер показателя	Значение показателя
1.2.1	Решение фрагмента программы одной из моделей алгоритмов
1.2.2	Выявление правильной модели построения алгоритмов из предложенных
1.2.3	Выявление определения из методов построения алгоритмов
1.2.4	Определение вида метода построения алгоритмов

### 2.3 Текущий контроль (ТК) № 3

**Тема занятия: (2.4.7.Составление алгоритмов на обработку двумерного массива. Сортировка элементов массива по заданному условию)**

Дидактические единицы	Основные показатели оценивания результата	№ задания относящийся к показателю оценивания	Метод контроля	Форма контроля	Вид контроля	Индексы занятий ранее изученных связанные с контролируемыми единицами
2.1	2.1.1, 2.1.2, 2.1.3, 2.1.4	3.1, 3.6, 3.2, 3.4, 3.5, 3.3, 3.7	Сравнение с аналогом	Практическая работа	Письменная практическая работа	2.1.2, 2.1.4, 2.1.5, 2.2.3, 2.2.4, 2.2.6, 2.3.3, 2.3.4, 2.3.5, 2.4.3, 2.4.4, 2.4.6

### Перечень заданий текущего контроля

Номер задания	Задания
3.1	<p>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.</p> <p><i>For i:=0 to 9 do</i></p> <p><i>A[i]:=9+I;</i></p> <p><i>For i:=0 to 4 do</i></p> <p><i>Begin</i></p> <p><i>K:=A[i]; A[i]:=A[9-i]; A[9-i]:=A[i]-k;</i></p> <p><i>End;</i></p> <p>Постройте блок-схему по программе. Чему будут равны элементы этого массива после выполнения фрагмента программы?</p> <p>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</p>

3.6 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 do Begin  
  
If (F(t)<r) then begin m:=t; r:=F(t); end;  
  
End;  
  
Write (m);  
  
End.
```

3.2 Определите значение переменной с после выполнения следующего фрагмента программы.

	<pre> a:=20; b:=30;  b:=-a-3*b;  If a&lt;b then c:=b-a else c:=a-2*b; </pre>
3.4	<p>Алгоритм вычисления значения функции <math>F(n)</math>, где <math>n</math> – натуральное число, задан следующими соотношениями:</p> $F(1) = 1,$ $F(n) = F(n-1) * n, \text{ при } n > 1.$ <p>Чему равно значение функции <math>F(6)</math>? В ответ запишите только натуральное число</p>
3.5	<p>Ниже представлена программа, которая получив на вход число <math>x</math>, печатает 2 числа <math>L</math> и <math>M</math>. Укажите наибольшее из таких чисел <math>x</math>, при вводе которых программа печатает сначала 2, а потом 8.</p> <pre> Var x, L, M: integer;  Begin  Readln (x);  L:=0; M:=0;  While x&gt;0 do begin  L:=L+1; </pre>

*If*  $M < (x \bmod 10)$  *then*  $M := x \bmod 10;$

$X := x \bmod 10;$  *end;*

*Writeln (L); writeln(M);*

*End.*

- 3.3 Определите, что будет напечатано в результате работы следующего фрагмента программы:

*Var k, s: integer;*

*Begin*

$S := 0;$   $k := 0;$

*While*  $s \leq 2048$  *do*

*Begin*

$S := s + 64;$   $k := k + 1;$

*End;*

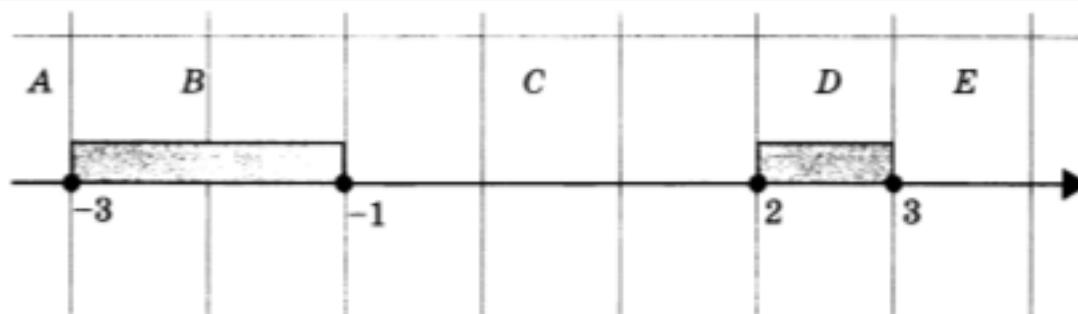
*Write (k);*

*End.*

Постройте блок-схему по данному фрагменту программы.

3.7

Требовалось написать программу, при выполнении которой с клавиатуры считывается координата точки на числовой прямой (действительное число  $x$ ) и определяется принадлежность этой точки заданным закрашенным областям (включая границы). Программист торопился и написал программу неправильно.



*Var x: real;*

*Begin*

*Readln(x);*

*If*  $x \leq 3$  *then*

*If*  $x \geq -3$  *then*

*If*  $x \geq 2$

*Write ('принадлежит')*

*Else write ('не принадлежит')*

*End.*

*Определите ошибку*

### **Перечень показателей текущего контроля**

Номер показателя	Значение показателя
2.1.1	Решение алгоритма для конкретной задачи
2.1.2	Определение результата выполнения алгоритма для конкретной задачи
2.1.3	Построение блок-схемы для конкретного алгоритма
2.1.4	Выявление ошибки в алгоритме для конкретной задачи

### **3. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ, ИСПОЛЬЗУЕМЫЙ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

№ семестра	Вид промежуточной аттестации
5	Экзамен

Дидактические единицы	Основные показатели оценивания	№ задания относящийся к показателю	Метод контроля	Форма контроля	Вид контроля	Индексы занятий ранее изученных связанные с

	результат	оценивания				контролируемыми дидактическими единицами
1.3	1.3.1	1	Опрос	Контрольная работа	Письменная контрольная работа	1.1.4, 2.2.5
2.1	2.1.1, 2.1.2, 2.1.3, 2.1.4	1, 1, 2, 1, 2, 1, 2	Информационно- аналитический	Контрольная работа	Письменная контрольная работа	2.1.2, 2.1.4, 2.1.5, 2.2.3, 2.2.4, 2.2.6, 2.3.3, 2.3.4, 2.3.5, 2.4.3, 2.4.4, 2.4.6, 2.4.7, 2.5.3, 2.5.4
2.2	2.2.1	1	Информационно- аналитический	Контрольная работа	Письменная контрольная работа	2.2.6

### Перечень заданий текущего контроля

Номер задания	Задания
1	<p><i>Составить для задачи блок-схему, разработать алгоритм, определить сложность работы данного алгоритма.</i></p> <p>Задача 1 вариант:</p> <p>Дано натуральное число. Определить сколько раз в нем встречается последняя цифра.</p>

Задача 2 вариант:

Дано натуральное число. Определить произведение его цифр, больших семи

Задача 3 вариант:

Дано натуральное число. Определить на сколько его максимальная цифра превышает минимальную

Задача 4 вариант:

Дано натуральное число. Определить количество цифр в числе

Задача 5 вариант:

Дано натуральное число. Определить количество всех четных чисел последовательности

2

***Разработать алгоритм, сделать реверс в строке по номеру варианта***

Задача 1 вариант:

Заполнить массив размером 6х6 так, как показано на рисунке

1	1	1	1	1	1
1	2	3	4	5	6
1	3	6	10	15	21
1	4	10	20	35	56
1	5	15	35	70	126
1	6	21	56	126	252

Задача 2 вариант:

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

Задача 3 вариант:

Заполнить массив так, как показано на рисунке

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

Задача 4 вариант:

Заполнить массив размером 6х6 так, как показано на рисунке

1	0	0	0	0	1
1	1	0	0	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	0	0	1	1
1	0	0	0	0	1

Задача 5 вариант:

Дан двумерный массив целых чисел размером 3x10, данные вводятся с клавиатуры. Определить:

- а) сумму нечетных элементов массива;
- б) количество элементов массива, больших 10

### **Перечень показателей текущего контроля**

Номер показателя	Значение показателя
1.3.1	Обоснование выбора и применения методов вычисления сложности работы алгоритмов
2.1.1	Построение блок-схемы для конкретного алгоритма
2.1.2	Определение метода алгоритма для разработки
2.1.3	Соблюдение синтаксиса при разработке алгоритма
2.1.4	Результативное создание алгоритма для конкретной задачи
2.2.1	Результативное определение сложности работы алгоритма