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

Методические указания по выполнению самостоятельной работы по дисциплине

ОП.08 Теория алгоритмов

специальности

09.02.03 Программирование в компьютерных системах

Иркутск, 2017

РАССМОТРЕНЫ
ПКС протокол № 12 от 19.05.2017г.
Председатель ЦК
_____ / М.А. Кудрявцева /

УТВЕРЖДАЮ
Зам. директора по УР
_____ Е.А. Коробкова

№	Разработчик ФИО
1	Еримеев Антон Михайлович

Пояснительная записка

Дисциплина ОП.08 Теория алгоритмов входит в Общепрофессиональный цикл. Самостоятельная работа является одним из видов внеаудиторной учебной работы обучающихся.

Основные цели самостоятельной работы:

- систематизация и закрепление теоретических знаний и практических умений, обучающихся
- углубление и расширение теоретических знаний, формирование умений использовать справочную документацию и дополнительную литературу;
- развитие познавательных способностей и активности обучающихся, творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельного мышления;
- развитие исследовательских умений.

Методические рекомендации помогут обучающимся целенаправленно изучать материал по теме, определять свой уровень знаний и умений при выполнении самостоятельной работы.

Рекомендации для обучающихся по выработке навыков самостоятельной работы:

- Слушать, записывать и запоминать лекцию.
- Внимательно читать план выполнения работы.
- Выбрать свой уровень подготовки задания.
- Обращать внимание на рекомендуемую литературу.
- Из перечня литературы выбирать ту, которая наиболее полно раскрывает вопрос задания.
- Учиться кратко излагать свои мысли.
- Использовать общие правила написания конспекта.
- Оценивать, насколько правильно понято содержание материала, для этого придумать вопрос, направленный на уяснение материала.
- Обращать внимание на достижение основной цели работы.

Тематический план

Раздел Тема	Тема занятия	Название работы	Количество часов
Раздел 1. Основные сведения об алгоритмах Тема 1. Основные понятия, свойства и способы	Введение в дисциплину. Понятие алгоритма. Типы алгоритмов, свойства.	Дать определение "детерминированность"	1
	Способы описания алгоритмов. Блок-схема	Привести примеры алгоритмов Евклида	1
	Основы Pascal. Операторы, функции, процедуры.	Подготовка презентации на тему "Основные понятия, свойства и способы описания алгоритмов"	1
	Оценка эффективности алгоритма.	Привести примеры проверки эффективности над счетчиком	1
Раздел 2. Построение алгоритмов Тема 1. Конструирование простейших алгоритмов	Линейные алгоритмы	Привести примеры линейных алгоритмов из жизни	1
	Решение задач на составление линейных алгоритмов	Проверка решенных задач в аудитории в среде ABC Pascal. Линейные алгоритмы.	1
	Разветвляющие алгоритмы	Привести примеры разветвляющихся алгоритмов из жизни	1
	Решение задач на составление разветвляющихся алгоритмов	Проверить решенные задачи в аудитории в среде ABC Pascal. Решение задач на составление разветвляющихся алгоритмов.	1
	Решение задач на составление разветвляющихся алгоритмов	Проверить решенные задачи в аудитории в среде ABC Pascal. Разветвляющие алгоритмы.	1
Тема 2. Конструирование циклических алгоритмов	Циклические алгоритмы. Цикл с параметром	Подбор задач, которые можно решить с помощью цикла с параметром	1
	Цикл с постусловием и с предусловием	Составление презентации на тему «Типы алгоритмов»	1
	Решение задач на составление циклических алгоритмов	Проверить решенные задачи в аудитории в среде ABC Pascal. Решение задач на составление циклических	1

		алгоритмов.	
	Решение задач на составление циклических алгоритмов.	Придумать 2 задачи на циклические алгоритмы	1
	Вложенные циклы. Вычисление сложности циклического алгоритма	Привести пример задачи, при которой вложенность создаст помеху обработки данных	1
	Решение задач на составление циклических алгоритмов. Вложенные циклы. Определение сложности циклических алгоритмов	Выполнение теста	1
Тема 3. Алгоритмы обработки одномерных массивов	Основные понятия об одномерном массиве. Поиск и замена элементов массива по заданному условию. Удаление и добавление элементов массива по заданному условию	Разработка блок-схемы для графического изображения основных правил существования массива	1
	Сортировка элементов одномерного массива по заданному условию	Сформулировать условия задачи по заданным алгоритмам	1
	Составление алгоритмов на обработку одномерного массива. Поиск и замена элементов массива по заданному условию.	Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку одномерного массива.	1
	Составление алгоритмов на обработку одномерного массива. Удаление и добавление элементов массива по заданному условию	Решение 2 задач на обработку одномерного массива	1
	Составление алгоритмов на обработку одномерного массива. Сортировка элементов массива по заданному условию	Проверить решенные задачи в аудитории в среде ABC Pascal. Сортировка элементов массива по заданному условию.	2
Тема 4. Алгоритмы обработки двумерных массивов	Основные понятия о двумерном массиве. Поиск и замена элементов массива по заданному условию. Удаление и добавление элементов массива по	Определение результата выполнения алгоритмов для заданных исходных данных	1

	заданному условию		
	Сортировка элементов двумерного массива по заданному условию	Решение 2 задач на обработку двумерного массива	1
	Составление алгоритмов на обработку двумерного массива. Поиск и замена элементов массива по заданному условию	Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку двумерного массива.	1
	Составление алгоритмов на обработку двумерного массива. Удаление и добавление элементов массива по заданному условию	Составление блок-схем по заданным программам	2
	Составление алгоритмов на обработку двумерного массива. Удаление и добавление элементов массива по заданному условию.	Проверить решенные задачи в аудитории в среде ABC Pascal. Поиск и замена элементов массива по заданному условию.	1
Тема 5. Алгоритмы обработки текстовых данных	Алгоритмы обработки текстовых данных	Рассмотреть понятие функция и процедура	1
	Алгоритмы обработки текстовых данных	Решение 2 задач на обработку текстовых данных	1
	Составление алгоритмов на обработку текстовых данных	Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку текстовых данных.	1
	Составление алгоритмов на обработку текстовых данных	Проверить решенные задачи в аудитории в среде ABC Pascal. Алгоритмы обработки текстовых данных.	1
	Смешанные алгоритмы	Подготовка эссе "Я и Теория алгоритмов"	1

Самостоятельная работа №1

Название работы: Дать определение "детерминированность".
Цель работы: углубление и расширение полученных теоретических знаний студентов.
Уровень СРС: воспроизводящая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
дать определение "детерминированность".
Образец ответа:
Детерминированность (от лат. determinans — определяющий) — определяемость. Детерминированность может подразумевать определяемость на общегносеологическом уровне или для конкретного алгоритма. Под жёсткой детерминированностью процессов в мире понимается однозначная предопределённость, т. е. у каждого следствия есть строго определённая причина. В таком смысле является антонимом стохастичности. Но детерминированность не всегда тождественна предопределённости. Например, может быть детерминированность будущим (целевая детерминация), когда предполагаемые субъектом цели в его возможном будущем определяют его поведение в настоящем.
Критерии оценки:
оценка «5» - Дано полное формальнологическое определение термину. Пояснено и приведены примеры применения термина.
оценка «4» - Дано полное формальнологическое определение термину. Приведены примеры.
оценка «3» - Дано определение термину.

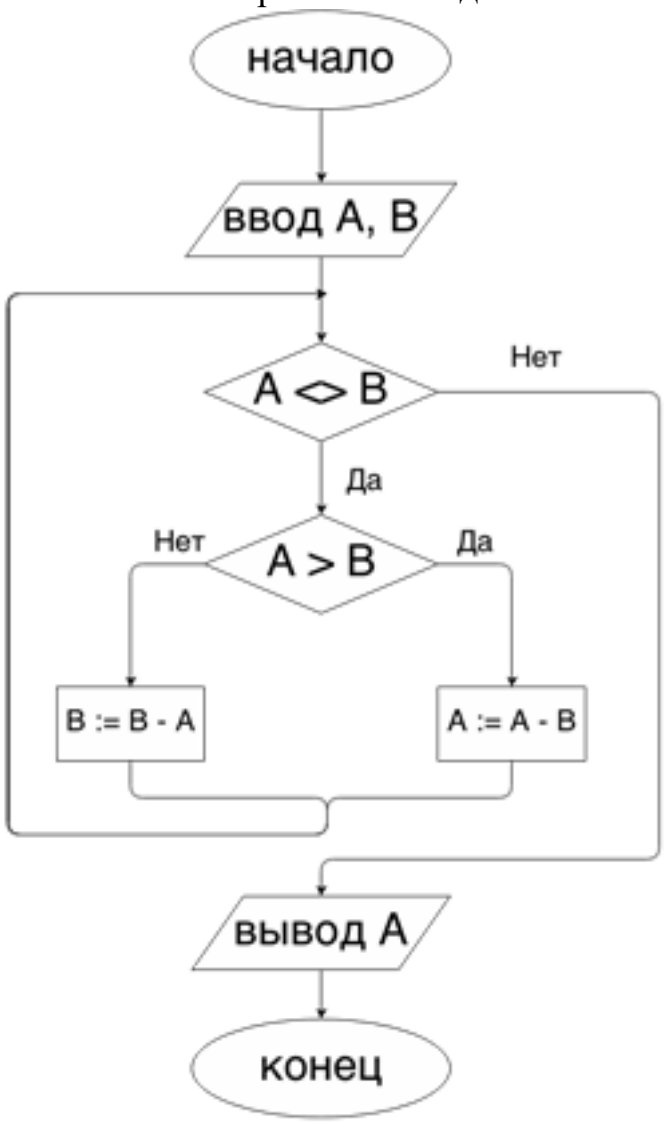
Самостоятельная работа №2

Название работы: Привести примеры алгоритмов Евклида.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: реконструктивная.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Привести примеры алгоритмов Евклида.
Например:

Алгоритм Евклида «с вычитанием»
Пусть a и b — целые числа, тогда верны следующие утверждения:
Все общие делители пары a и b являются также общими делителями пары a — b, b;
И наоборот, все общие делители пары a — b и b являются также общими

делителями пары a и b;
НОД(A, B) = НОД(A — B, B) , если A > B;
НОД(A, 0) = A.

Блок-схема алгоритма Евклида «с вычитанием»



Листинг:

```
var
a, b: integer;

begin
write('a = ');
readln(a);
write('b = ');
readln(b);
while a <> b do
if a > b then
```

```
a := a - b
else
b := b - a;
writeln('NOD = ', a);
end.
```

Критерии оценки:

оценка «5» - Приведены три примера алгоритма Евклида
оценка «4» - Приведены два примера алгоритма Евклида
оценка «3» - Приведен один пример алгоритма Евклида

Самостоятельная работа №3

Название работы: Подготовка презентации на тему "Основные понятия, свойства и способы описания алгоритмов".

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: отчет в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

подготовить презентацию на тему "Основные понятия, свойства и способы описания алгоритмов"

Критерии оценки:

оценка «5» - Презентация содержит 25 слайдов, оформлена без замечаний.
Тема раскрыта полно.
оценка «4» - Презентация содержит 20-25 слайдов. Презентация оформлена с замечаниями: не более двух замечаний.
Тема раскрыта частично: не более двух замечаний.
оценка «3» - Презентация содержит менее 20 слайдов. Презентация оформлена с замечаниями: не более четырех замечаний.
Тема раскрыта частично: не более четырех замечаний.

Самостоятельная работа №4

Название работы: Привести примеры проверки эффективности над счетчиком.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

Выведите НОД n чисел - Pascal.

Листинг:

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

```
m:array[1..1000] of integer; function nod(a,b:longint):longint; begin
if b mod a=0 then nod:=a else nod:=nod(b,a mod b); end;
begin write('n='); read(n); label m1; i:=1;
read(m[i]);
i:=i+1;
if i<n then goto m1; g:=m[1];
label m2; k:=2
g:=nod(g,m[i]); k:=k+1;
if k<n then goto m2; writeln('NOD=',g); readln;
end.
```

Критерии оценки:

оценка «5» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.

Самостоятельная работа №5

Название работы: Привести примеры линейных алгоритмов из жизни.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: реконструктивная.

Форма контроля: Письменный отчет в рабочей тетради.

Количество часов на выполнение: 1 час.

Задание:

привести примеры линейных алгоритмов из жизни.

Например: алгоритм приготовления быстро завариваемой лапши.

1. Раскрываем пакетик
2. Вываливаем лапшу в тарелку

3. Добавляем специи
4. Завариваем лапшу

Если не следовать алгоритму, то нет смысла его использовать. Вначале выполним пункт 2 и 3, а затем 1 и 4. Что получится?

Нарушая пункт 1 переходим к пункту 2, но тут проблема, не можем вывалить лапшу, ведь мы не открыли пакетик. Здесь идет явное нарушение алгоритма и дальнейшее его использование не валидно.

Критерии оценки:

оценка «5» - приведено три примера
оценка «4» - приведено два примера
оценка «3» - приведен один пример

Самостоятельная работа №6

Название работы: Проверка решенных задач в аудитории в среде ABC Pascal. Линейные алгоритмы..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: реконструктивная.

Форма контроля: отчет на бумажном носителе и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

Задание: решить задачи согласно выданному варианту.

1.Двое рабочих заработали вместе 900 рублей. Один работал 2 недели, а другой 4 недели. Сколько денег заработал каждый?

2.Составите программу для вычисления площади боковой поверхности цилиндра ($S= 2 \pi R h$).

3.Составите программу для вычисления площадь поверхности шара ($S = \pi D2$).

4.Найти разность, сумму и произведение трёх значений переменных.

5.Даны два действительных числа. Найти среднее арифметическое этих чисел и среднее геометрическое их модулей. Результат округлить до 2-х знаков после запятой.

6.Смешано V1 литров воды температуры t1 с V2 литрами воды температуры t2. Найти объём и температуру образовавшейся смеси.

7.Найти катет прямоугольного треугольника, если известна гипотенуза и прилежащий к искомому катету угол. Результат округлить до 2-х знаков после запятой.

8.Определить время падения камня на поверхности земли с высоты h.

9.Дана сторона равностороннего треугольника. Найти площадь этого треугольника. Результат округлить до 2-х знаков после запятой.

10.Определить силу притяжения F между телами массы m1 и m2 , находящимся на расстоянии r друг от друга. Результат округлить до 2-х знаков после запятой.

Критерии оценки:

оценка «5» - Решено десять задач, в решении присутствует:
Описание входных данных (их типов, диапазонов).
Описание выходных данных.
Запись математического соотношения, связывающего результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».
оценка «4» - Решено пять задач, в решении присутствует:
Описание входных данных (их типов, диапазонов).
Описание выходных данных.
Запись математического соотношения, связывающего результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.
оценка «3» - Решено три задачи, в решении присутствует:
Описание входных данных (их типов, диапазонов).
Описание выходных данных.
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №7

Название работы: Привести примеры разветвляющихся алгоритмов из жизни.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради.
Количество часов на выполнение: 1 час.
Задание:
Привести примеры разветвляющихся алгоритмов из жизни.
Например:
Посмотрел в окно.
Идет дождь - Остаться дома. Не идет дождь - Пойти гулять.
Критерии оценки:
оценка «5» - Приведено три примера.
оценка «4» - Приведено два примера.
оценка «3» - Приведен один пример.

Самостоятельная работа №8

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Решение задач на составление разветвляющихся алгоритмов..
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: реконструктивная.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Листинг:

1. program parallelipiped; var a,b,c: integer;
var V,S: real; begin
a:=6;
b:=3;
c:=9;
V:=a*b*c; S:=2*(a*b+b*c+a*c); writeln (V);
writeln (S); end.
2. program formula; var a,b,c,d,m,n: integer; begin
a:=9;
b:=3;
c:=5;
d:=2;
m:=a+(b*c-d); n:=d-a*2;
write (m);
write (n); write (m/n); end.
3. program srednee arefmetichescoe; var a,b,c,d: integer;
begin a:=8;
b:=3;
c:=5;
writeln ((a+b+c)/3); end.
4. Program Myl_1 ; Var a, b, r : Integer; Begin
WriteLn ('Введите два числа через пробел');
ReadLn (a, b); r:=a*b;
WriteLn ('Их произведение равно ', r); WriteLn ('Нажмите <Enter>');
End.
5. Program Okr; Var p, r : real; Begin
WriteLn ('Введите радиус'); ReadLn (r);
p:=2*3.14*r;
Write ('Длинна окружности = ', p); ReadLn
End.
6. Program Kub; var a: integer; S1, S2, V: real; begin
readln (a); S1:= sqr(a);
S2:= 6*sqr(a);
V:=sqr(a)*a;
writeln ('Площадь грани куба = ', S1); writeln ('Площадь полной поверхности

куба = ', S2);
writeln ('Объём куба = ', V); end.

7. Program formula; var a, b, c: integer; D: real;
begin
readln (a,b,c); D:=(sqr(b)+sqrt(c))/abs(a); writeln ('Формула = ', D) end.

8. Program summa; var n, k: integer;
A: real; begin readln (n,k);
A:=(1200*n)+(3800*k);
writeln ('Мальчик потратил ', A) end.

9. Program formula_1; var a, b, c: integer;
S: real; begin
readln (a,b,c); S:=((a+b)/2)*(sqrt(sqr(c)-sqr(a-b)/4)); writeln ('Формула = ', S)
end.

10. Program formula_1; var r, k: integer;
S: real; begin
readln (r, k); S:=3,14*(sqr(r)-sqr(k)); writeln ('Формула = ', S) end.

Критерии оценки:

оценка «5» - Решено десять задач (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического соотношения, связывающего результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено пять задач (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического соотношения, связывающего результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решенотри задачи (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №9

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Разветвляющие алгоритмы..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: реконструктивная.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Листинг:

1. Program Kub; var a: integer; S1, S2, V: real; begin
readln (a); S1:= sqr(a);
S2:= 6*sqr(a);
V:=sqr(a)*a;
writeln ('Площадь грани куба = ', S1); writeln ('Площадь полной поверхности куба = ', S2);
writeln ('Объём куба = ', V); end.
2. Program formula; var a, b, c: integer; D: real;
begin
readln (a,b,c); D:=(sqr(b)+sqrt(c))/abs(a); writeln ('Формула = ', D) end.
3. Program summa; var n, k: integer;
A: real; begin readln (n,k);
A:=(1200*n)+(3800*k);
writeln ('Мальчик потратил ', A) end.
4. Program formula_1; var a, b, c: integer;
S: real; begin
readln (a,b,c); S:=((a+b)/2)*(sqrt(sqr(c)-sqr(a-b)/4)); writeln ('Формула = ', S) end.
5. Program formula_1; var r, k: integer;
S: real; begin
readln (r, k); S:=3,14*(sqr(r)-sqr(k)); writeln ('Формула = ', S) end.

Критерии оценки:

оценка «5» - Решено пять задач (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического соотношения, связывающего результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического соотношения, связывающего результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов).Описание выходных данных.Запись математического

соотношения, связывающего результат с исходными данными.

Самостоятельная работа №10

Название работы: Подбор задач, которые можно решить с помощью цикла с параметром.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

Найти в глобальной сети три задачи, которые можно решить с помощью цикла с параметром.

Критерии оценки:

оценка «5» - Приведено три примера.
Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов про- грамм, данных и систем»;

оценка «4» - Приведено два примера. Необходимо выполнить следующие 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.

оценка «3» - Приведен один пример. Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №11

Название работы: Составление презентации на тему «Типы алгоритмов».

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: проверка выполненной работы в электронном виде.

Количество часов на выполнение: 1 час.

Задание:
Составить презентацию по теме «Типы алгоритмов».
Критерии оценки:
оценка «5» - Презентация содержит 25 слайдов, оформлена без замечаний.
Тема раскрыта полно.
оценка «4» - Презентация содержит 20-25 слайдов. Презентация оформлена с замечаниями: не более двух замечаний.
Тема раскрыта частично: не более двух замечаний.
оценка «3» - Презентация содержит менее 20 слайдов. Презентация оформлена с замечаниями: не более четырех замечаний.
Тема раскрыта частично: не более четырех замечаний.

Самостоятельная работа №12

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Решение задач на составление циклических алгоритмов..
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: реконструктивная.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Листинг:

```
1. Program Kub; var a: integer; S1, S2, V: real; begin
readln (a); S1:= sqr(a);
S2:= 6*sqr(a);
V:=sqr(a)*a;
writeln ('Площадь грани куба = ', S1); writeln ('Площадь полной поверхности
куба = ', S2);
writeln ('Объём куба = ', V); end.
2. Program formula; var a, b, c: integer; D: real;
begin
readln (a,b,c); D:=(sqr(b)+sqrt(c))/abs(a); writeln ('Формула = ', D) end.
3. Program summa; var n, k: integer;
A: real; begin readln (n,k);
A:=(1200*n)+(3800*k);
writeln ('Мальчик потратил  ', A) end.
4. Program formula_1; var a, b, c: integer;
S: real; begin
readln (a,b,c); S:=((a+b)/2)*(sqrt(sqr(c)-sqr(a-b)/4)); writeln ('Формула = ', S)
end.
5. Program formula_1; var r, k: integer;
```

S: real; begin
readln (r, k); S:=3,14*(sqr(r)-sqr(k)); writeln ('Формула = ', S) end.

Критерии оценки:

оценка «5» - Решено пять задач (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию; Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №13

Название работы: Придумать 2 задачи на циклические алгоритмы.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

Придумать и представить решение двух задачи на циклические алгоритмы.

Критерии оценки:

оценка «5» - Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»;

оценка «4» - Необходимо выполнить следующие 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.

оценка «3» - Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №14

Название работы: Привести пример задачи, при которой вложенность создаст помеху обработки данных.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Привести пример задачи, при которой вложенность создаст помеху обработки данных.
Пример:
Напечатать последовательность чисел в обратном порядке, используя рекурсивный вызов процедуры.
Например:
row (5) = 5 4 3 2 1
Из условия задачи ясно, что условием завершения рекурсии будет сам аргумент функции, который следует уменьшать на единицу, пока он >= 1.
Листинг:

```
procedure row(n:integer);
begin
if n >=1 then begin write (n, ' '); row(n-1)
end; end; begin
row(10); end.
```

Пример:
Написать рекурсивную процедуру, выводящую цифры, переданного ей в качестве фактического параметра числа, в обратном порядке.
Например: при переданном функции числе 3078, должно в итоге вернуть 8703
Использовать операции div и mod.
Листинг:

```
procedure reverse (n: integer); begin
write (n mod 10);
```

```
if (n div 10) <> 0 then reverse(n div 10)
end; begin writeln;
reverse(3078);
end.
```

Критерии оценки:

оценка «5» - Приведены три примера.
Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Приведено три примера.
Необходимо выполнить следующие 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.

оценка «3» - Приведено три примера.
Необходимо выполнить следующие приемы:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №15

Название работы: Выполнение теста.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: проверка решения теста в рабочей тетради.

Количество часов на выполнение: 1 час.

Задание:

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 счёта с помощью цифр, но поначалу слово «цифра» относилось только к нулю

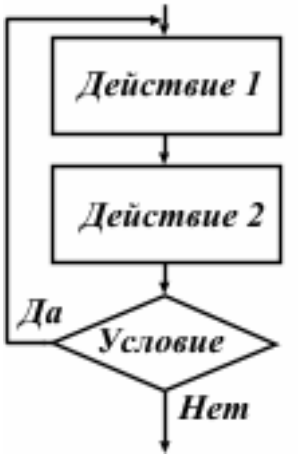
2. Какое из перечисленных свойств не относится к свойствам алгоритмов?
Дискретность Определенность Результативность Полнота Массовость Правильность

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

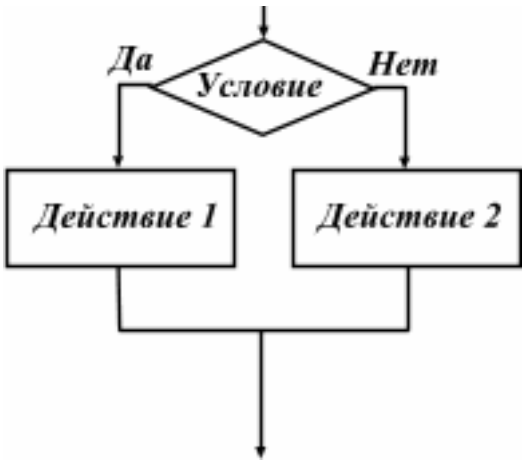
5.Какое свойство алгоритмов характеризует данное определение: «Алгоритм решения задачи разрабатывается в общем виде, т.е. он должен быть приме- ним для некоторого класса задач, различающихся лишь исходными данными»
Дискретность Определенность Результативность Полнота Массовость Правильность

6. Какой вид алгоритма представлен на рисунке?



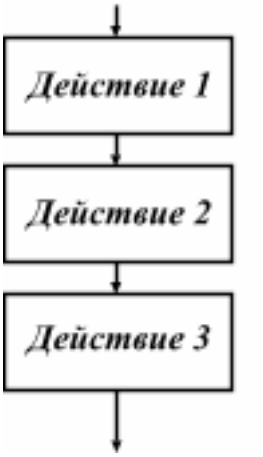
Линейный Разветвляющий Циклический

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»
детерминированным результативным объективным полным

Критерии оценки:

оценка «5» - Обучающийся ответил на 18-20 вопросов
оценка «4» - Обучающийся ответил на 14-17 вопросов
оценка «3» - Обучающийся ответил на 6-13 вопросов

Самостоятельная работа №16

Название работы: Разработка блок-схемы для графического изображения основных правил существования массива.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: воспроизводящая.

Форма контроля: письменный отчет в рабочей тетради .

Количество часов на выполнение: 1 час.

Задание:

Разработать блок-схему для графического изображения основных правил существования трехмерного массива.

Критерии оценки:

оценка «5» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».
оценка «4» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.

Записано математическое соотношение, связывающее результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.

Самостоятельная работа №17

Название работы: Сформулировать условия задачи по заданным алгоритмам.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов .
Уровень СРС: воспроизводящая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
Листинг:

```
Program posled;
Var a: array[1..100] of integer; i, n: integer;
Begin
Write (‘Сколько элементов? ’); Readln (n);
For i=1 to n do
begin
a[i]:= Random(58)-23;
writeln (a[i],’ ‘);
end; End.
```

Сформировать и вывести на экран последовательность из n элементов, заданных датчиком случайных чисел на интервале [-23, 34].
Задание:

- Листинг: Program proisveden;

```
Var a: array[1..100] of integer; i, n, p: integer;
Begin
Write (‘Сколько элементов? ’); Readln (n); p:=1;
For i:=1 to n do begin
write (‘введите число’); readln (a[i]);

p:=p*a[i]; end;
writeln(‘произведение элементов равно: ‘,p); End.
```

- Листинг: Program summa;

```
Var a: array[1..100] of real; i, n: integer;
s: real; Begin
Write (‘n=’); Readln (n); s:=0;
For i:=1 to n do begin
write (‘введите число’); readln (a[i]); s:=s+a[i];
end;
writeln(‘сумма элементов равна ‘,s); End.
```

- Листинг:

```
Program srednee;
Var a: array[1..100] of real; i, n: integer;
s,sred: real; Begin
Write (‘n=’); Readln (n); s:=0;
For i:=1 to n do begin
write (‘введите число’); readln (a[i]); s:=s+a[i];
end; sred:=s/n;
writeln(‘среднее арифметическое элементов: ‘,s); End.
```

Критерии оценки:

оценка «5» - Выполнено три задания.
Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Выполнено два задания.
Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Выполнено одно задание.
Описаны входные данные (их типы, диапазон).
Описаные выходные данные.
Записано математическое соотношение, связывающее результат с исходными данными.

Самостоятельная работа №18

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку одномерного массива..

Цель работы: систематизация, закрепление, углубление теоретических знаний

студентов.
Уровень СРС: эвристическая.
Форма контроля: отчет на бумажном носителе и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
1. Ввести число. Если оно неотрицательно, вычесть из него 10, в противном случае прибавить к нему 10.
2. Ввести два числа. Если их произведение отрицательно, умножить его на -2 и вывести на экран, в противном случае увеличить его в 3 раза и вывести на экран.
3. Ввести два числа. Если сумма этих чисел четная, найти произведение, в противном случае, найти частное этих чисел.
4. Ввести два числа. Вычесть из большего меньшее.
5. Ввести число. Если оно больше 10, разделить его на 2, если меньше или равно 10, то умножить на 5.
6. Ввести два числа. Если их сумма больше 100, то сумму уменьшить в 2 раза, в противном случае увеличить в 2 раза.
7. Ввести двухзначное число. Если сумма цифр числа четная, то увеличить число на 2, в противном случае уменьшить на 2.
8. Составить программу, которая по трем введенным вами числами определит, могут ли эти числа быть длинами сторон треугольника.
9. Дано целое число. Если оно является положительным, то прибавить к нему 20, в противном случае вычесть из него 5. Вывести полученное число (Написать программу, начертить блок-схему в тетради).
10. Дано два числа. Если их сумма кратна 5, прибавить 1, иначе вычесть 2.
Критерии оценки:
оценка «5» - Решено десять задач (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию. Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».
оценка «4» - Решено пять задач (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.
оценка «3» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит: Необходимо выполнить следующие задачи набрав в среде ABC Pascal.Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения,

связывающего результат с исходными данными.

Самостоятельная работа №19

Название работы: Решение 2 задач на обработку одномерного массива.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:

1. В заданном одномерном массиве, состоящем из n целых чисел, подсчитать количество нулей.
2. В заданном одномерном массиве, состоящем из n целых чисел, подсчитать количество четных элементов.

Критерии оценки:

оценка «5» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №20

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Сортировка элементов массива по заданному условию..
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов..
Уровень СРС: эвристическая.
Форма контроля: Письменный отчет в рабочей тетради и в электронном виде..

Количество часов на выполнение: 2 часа.

Задание:

1. Найти сумму элементов массива с четными номерами, содержащего N элементов. Элементы вводятся с клавиатуры.
2. Сформировать и вывести на экран массив, элементы которого заданы датчиком случайных чисел на интервале [-19, 26] (размер произвольный). Найти произведение элементов с нечетными номерами.
3. Сформировать и вывести на экран массив, элементы которого заданы датчиком случайных чисел на интервале [-56, 47] (размер произвольный). Найти произведение элементов с четными номерами, которые превосходят некоторое число t.
4. Найти наименьший элемент одномерного массива, состоящего из n элементов. Элементы вводятся с клавиатуры.
5. Найти номер наименьшего элемента в массиве, заданного датчиком случайных чисел на интервале [-20, 25]. Размер произвольный.

Критерии оценки:

оценка «5» - Решено пять задач (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию. Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №21

Название работы: Определение результата выполнения алгоритмов для заданных исходных данных.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:
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.
Критерии оценки:
оценка «5» - Обучающийся должен:

1. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов.
2. Описать каждый метод:

- 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)
- 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.

оценка «4» - Обучающийся должен

1. Указать в виде оценивающего ресурса процессорное время (*вычислительная сложность*) и память (*сложность алгоритма по памяти*). Оценка позволяет предсказать время выполнения и сравнивать эффективность алгоритмов.
2. Описать один из двух методов:

- 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)

- 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.

оценка «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.

Самостоятельная работа №22

Название работы: Решение 2 задач на обработку двумерного массива.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:

1. Сформировать с помощью датчика случайных чисел и вывести на экран матрицу, размером MxN. Элементы задаются на интервале [-20, 25].
2. В двумерном массиве, состоящем из n целых чисел, найти сумму элементов в каждой строке. Размер произвольный.

Критерии оценки:

оценка «5» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию.
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:

Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Задачи решены (набраны в среде ABC Pascal). Решение содержит:
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №23

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку двумерного массива..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

1. Найдите среднее арифметическое элементов массива, состоящего из 10 чисел, которые превышают по величине число С. Элементы вводятся с клавиатуры.

```
 Program sredarifmet;
 Var a: array[1..10] of real; i, k: integer;
 C, S, sred: real; Begin
 For i:=1 to 10 do begin
 write(‘a[‘,i,’]=’); readln (a[i]); end;
 write(‘введите С: ‘); readln (C );
For i:=1 to 10 do begin
 If a[i]>C then
 begin S=S+a[i]; K=K+1; end; end;
 sred=S/k;
 Writeln(‘среднее арифметическое чисел, превосходящих ‘,C,’ равно ‘,sred); End.
```

2.Найти произведение элементов целочисленного одномерного массива с четными номерами, состоящего из n элементов. Элементы вводятся с клавиатуры.

```
 Program proizved_chet;
 Var a: array [1..100] of integer; i, n, p: integer;
 Begin p:=1;
 write ('n='); readln (n); for i:=1 to n do
 begin
 write ('a[',i,']='); readln (a[i]); if i mod 2=0 then p:=p*a[i];
```

```
 end;
 Writeln ('произведение элементов массива с четными номерами равно ',p);
 End.
```

3.Массив А вводится с клавиатуры. Сформировать новый массив В, состоящий из четных элементов массива А. Элементы вводятся с клавиатуры. Размер n.

Kol – количество элементов в последовательности В К – номер элемента последовательности В

```
 Program newmasiv;
 Var a: array[1..100] of integer; b: array[1..100] of integer;
 kol, n, i, k: integer; Begin
 kol:=0; k:=0;
 write ('n='); readln (n); For i:=1 to n do
 begin
 write('a[',i,']='); readln (a[i]); if a[i] mod 2=0 then
 begin
k:=k+1; b[k]:=a[i]; kol:=kol+1; end;
 end;
 if kol=0 then writeln('четных элементов нет') else for k:=1 to kol do
write('b[',k,']=',b[k]);
 end;
```

Критерии оценки:

оценка «5» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию; Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»;

оценка «4» - Решено две задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №24

Название работы: Составление блок-схем по заданным программам.

Цель работы: систематизация, закрепление, углубление теоретических знаний

студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 2 часа.
Задание:

1. Подсчитать количество положительных элементов в каждой строке матрицы размером MxN, элементы которой вводятся с клавиатуры.

Листинг:

```
program kolpolvstr;
Var a: array[1..50,1..50] of inte- ger; i, j, m, n, kol: integer;
Begin
Write(‘сколько строк?’); Readln(m); Write(‘сколько столбцов?’);
Readln(n); For i:=1 to m do
begin
For j:=1 to n do
begin write(‘a[‘,i,’,’,j,’]=’); readln (a[i,j]); end; for i:=1 to m do
begin kol:=0;
for j:=1 to n do if a[i,j]>0 then kol:=kol+1;
writeln('количество положительных элементов в ',i,' строке: ',kol); end;
writeln; end;
```

- end.

2. Вывести на экран произведение двузначных элементов последовательности натуральных чисел, которые делятся на заданное число.

Листинг:

```
Program_1; var
a, n, prod: word; begin
readln(n); read(a); prod := 1;
while a <> 0 do begin
if (a > 0) and (a < 100) and (a mod n = 0) then prod := prod * a;
read(a) end;
if prod <> 1 then writeln(prod) else writeln(' No such elements!') end.
```

3. Найти количество простых членов последовательности

```
program NumOfPrimes;
var
a, i, s, count: word; begin
read(a); count := 0;
while a <> 0 do begin s := 0;
for i := 1 to a do begin
if a mod i = 0 then inc(s) end;
if s = 2 then inc(count); read(a)
end;
writeln(count)
```

end.

Критерии оценки:

оценка «5» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию;
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем»;

оценка «4» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №25

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Поиск и замена элементов массива по заданному условию..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

- Найти наименьший элемент двумерного массива. Размер MXN. Элементы задаются на интервале [-30, 45].

Листинг:

```
Program minim;
 Var a: array[1..50,1..50] of integer; i, j, n, m, min: integer;
 Begin
 Write('сколько строк?'); Readln(m); Write('сколько столбцов?'); Readln(n); For i:=1
```

```
to m do
 begin
 For j:=1 to n do begin
 a[i,j]:=int(rnd*76)-30; write(a[i,j],' '); end;
 writeln;
 end;
 min:=a[1,1];
 For i:=1 to m do For j:=1 to n do
 if a[i,j]< min then min:=a[i,j]; Writeln('наименьшее число ',min);
 End.
```

- В двумерном массиве, состоящем из целых чисел, найти наименьший элемент и номер строки, в которой он находится. Элементы вводятся с клавиатуры. Размер MXN.

Листинг:

```
Program minim;
 Var a: array[1..50,1..50] of integer; i, j, m, n, min, K: integer;
 Begin
 Write('сколько строк?'); Readln(m); Write('сколько столбцов?'); Readln(n); For i:=1
to m do
 For j:=1 to n do
 begin write('a[',i,',',j,']='); readln (a[i,j]); end; min:=a[1,1]; K:=1;
 For i:=1 to m do For j:=1 to n do
 If a[i,j]< min then begin
 min:=a[i,j];    K:=i;
end;
 Writeln('наименьшее число  ',min,' находится в ', k , ' строке');
 End.
```

- Найти сумму элементов в каждой строке двумерного массива, состоящего из целых чисел. Размер MXN. Элементы задаются на интервале [-19, 30].

Листинг:

```
program sumstr;
 Var a: array[1..50,1..50] of integer; i, j, n, m,sum: integer;
 Begin
 Write('сколько строк?'); Readln(m); Write('сколько столбцов?'); Readln(n); For i:=1
to m do
 begin
 For j:=1 to n do begin
 a[i,j]:=int(rnd*50)-19; write(a[i,j],' '); end;
 writeln;
```

```
 end;
 for i:=1 to m do begin
 sum:=0;
 for j:=1 to n do sum:=sum+a[i,j]; writeln('сумма элементов в ',i,' строке: ',sum);
 end;
 end.
```

Критерии оценки:

оценка «5» - Решено три задач (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию; Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено две задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №26

Название работы: Рассмотреть понятие функция и процедура.

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради .

Количество часов на выполнение: 1 час.

Задание:

Конспект: Процедуры и функции в Pascal.

Критерии оценки:

оценка «5» - конспект составлен по плану, соблюдается логичность, последовательность изложения материала, качественное внешнее оформление, объем - 4 тетрадные страницы

оценка «4» - конспект выполнен по плану, но некоторые вопросы раскрыты не полностью, есть небольшие недочеты в работе, объем – 4 тетрадные страницы

оценка «3» - при выполнении конспекта наблюдается отклонение от плана, нарушена логичность, отсутствует внутренняя логика изложения, удовлетворительное
внешнее оформление, объем менее 4 страниц

Самостоятельная работа №27

Название работы: Решение 2 задач на обработку текстовых данных.
Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.
Уровень СРС: эвристическая.
Форма контроля: письменный отчет в рабочей тетради и в электронном виде.
Количество часов на выполнение: 1 час.
Задание:
1. Посчитать количество слов в предложении. Листинг:

```
Program KolSlov;
Var Text  : String; i, k : Integer; Flag: Boolean; BEGIN
WriteLn('Введите текст :');   ReadLn(Text); k := 0; Flag := TRUE;
For i := 1 to Length(Text) do begin
If (Text[i] <> ' ') and Flag then k := k+1;
Flag := (Text[i]=' ') end;
WriteLn('О т в е т : количество слов в тексте равно ', k); END.
```

2. Заменить слово в предложении
Программа, использующая стандартную функцию Pos , не требует, чтобы длины заменяемого и вставляемого слов были одинаковыми.
Листинг:

```
Program Replace;
Var Text, Slovo1, Slovo2 : String; i, DlinaSlova, P : Integer; BEGIN;
Write('Введите строку : '); ReadLn(Text); Write('Какое слово заменить ? ');
ReadLn(Slovo1);
Write('На какое слово заменить ? '); ReadLn(Slovo2); WriteLn;
DlinaSlova:=Length(Slovo1);
P:=Pos(Slovo1,Text); {номер позиции, с которой в строке Text в первый раз встречается подстрока Slovo1}
While P>0 do {цикл продолжается до тех пор пока подстрока Slovo1 встречается в строке Text}
begin Delete(Text, P, DlinaSlova); {удаление подстроки Slovo1, начинающейся с позиции P, из строки Text }
Insert(Slovo2, Text, P); {вставка подстроки Slovo2 в строку Text с позиции P}
P:=Pos(Slovo1, Text); {номер позиции, с которой подстрока Slovo1 встречается в строке Text в очередной раз} end;
WriteLn('Новый текст: ', Text);
END.
```

Критерии оценки:

оценка «5» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Алгоритм решения задачи соответствует математическому понятию;
Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными;
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Необходимо выполнить следующие задачи набрав в среде ABC Pascal.
Описание входных данных (их типов, диапазонов);
Описание выходных данных;
Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №28

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Составление алгоритмов на обработку текстовых данных..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

- Найти номер столбца массива размером MxN, в котором находится наибольшее количество отрицательных элементов. Элементы вводятся с клавиатуры.

Листинг:

```
program nomerstolb;
Var a: array[1..50,1..50] of integer; b: array[1..50] of integer;
i, j, m, n, max, jmax: integer; Begin
Write('сколько строк?'); Readln(m); Write('сколько столбцов?'); Readln(n); For i:=1
```

```
to m do
 begin
 For j:=1 to n do begin
 write('a[',i,',',j,']='); readln (a[i,j]);
 end;
 for j:=1 to m do begin
 b[j]:=0;
 for i:=1 to n do
 if a[i,j]<0 then b[j]:=b[j]+1;
end;
 max:=b[1]; jmax:=1; For j:=2 to n do
 begin
 if b[j]>max then begin
 max:=b[j]; jmax:=j; end;
 end;
 writeln('Наибольшее количество отрицательных элементов в ';jmax ; 'столбце');
end;
 end.
```

- Подсчитать количество положительных элементов в каждом столбце матрицы размером MxN, элементы которой вводятся с клавиатуры.
- Подсчитать количество отрицательных элементов в каждой строке матрицы размером MxN, элементы которой задаются с помощью датчика случайных чисел на интервале [-35; 65].

Критерии оценки:

оценка «5» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию; Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено две задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №29

Название работы: Проверить решенные задачи в аудитории в среде ABC Pascal. Алгоритмы обработки текстовых данных..

Цель работы: систематизация, закрепление, углубление теоретических знаний студентов.

Уровень СРС: эвристическая.

Форма контроля: письменный отчет в рабочей тетради и в электронном виде.

Количество часов на выполнение: 1 час.

Задание:

Упорядочить каждый столбец матрицы по возрастанию. Массив размером MxN, элементы которого задаются датчиком случайных чисел на интервале [-17;26].

Листинг: program porydok;

```
Var a: array[1..50,1..50] of integer; i, j, n, m,t,r: integer;
Begin
Write('сколько строк?'); Readln(m); Write('сколько столбцов?'); Readln(n);
For i:=1 to m do
begin
For j:=1 to n do begin
a[i,j]:=int(rnd*44)-17;
write(a[i,j],' ');
end; Writeln;
end;
For j:=1 to n do For r:=1 to m do
For i:=1 to m-1 do
if a[i,j]> a[i+1,j] then begin
t:= a[i,j]; a[i,j]:= a[i+1,j];
a[i+1,j]:= t;
end;
For i:=1 to m do begin
For j:=1 to n do write(a[i,j],' '); Writln;
end; End.
```

2. Подсчитать количество четных элементов в каждом столбце матрицы разме- ром MxN, элементы которой задаются с помощью датчика случайных чисел на интервале [-98; 54].

3. Подсчитать количество четных отрицательных элементов в матрице размером MxN, элементы которой вводятся с клавиатуры.

а. Сформировать матрицу **1 1 1 1**

2 2 2 2
3 3 3 3
4 4 4 4

б. Сформировать матрицу **1 2 3 4**
1 2 3 4
1 2 3 4
1 2 3 4

в. Сформировать матрицу **0 0 0 1**
0 0 1 0
0 1 0 0
1 0 0 0

Критерии оценки:

оценка «5» - Решено три задачи (набраны в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными; Алгоритм решения задачи соответствует математическому понятию; Блок-схема построена в соответствии с ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем».

оценка «4» - Решено две задачи (набраны в среде ABC Pascal). Решение содержит: Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными;
Блок-схема соответствует выбранному алгоритму решения задачи.

оценка «3» - Решена одна задача (набрана в среде ABC Pascal). Решение содержит:Описание входных данных (их типов, диапазонов);Описание выходных данных;Запись математического соотношения, связывающего результат с исходными данными.

Самостоятельная работа №30

Название работы: Подготовка эссе "Я и Теория алгоритмов".
Цель работы: развитие творческих способностей.
Уровень СРС: творческая.
Форма контроля: письменный отчет в рабочей тетради.
Количество часов на выполнение: 1 час.
Задание:
Подготовить эссе "Я и Теория алгоритмов"**.**
Выполнение задания:
1) написать вступление (2–3 предложения, которые служат для последующей формулировки проблемы).
2) сформулировать проблему, которая должна быть важна не только для автора, но и для других;
3) дать комментарии к проблеме;
4) сформулировать авторское мнение и привести аргументацию;

5) написать заключение (вывод, обобщение сказанного).

Критерии оценки:

оценка «5» - все аспекты, указанные в задании, раскрыты полностью, работа оформлена чисто, аккуратно и без исправлений, объём не менее 20 предложений.

оценка «4» - аспекты, указанные в задании, раскрыты кратко, структура соблюдена неточно (нет вступления и/или заключения). Объём не менее 15 предложений.

оценка «3» - задание выполнено частично: раскрыто только два аспекта, указанные в задании. Эссе написано, но структура не соблюдена. Объём менее 10 предложений.