

БПОУ ВО «Губернаторский колледж народных промыслов»

**Комплект
контрольно-оценочных средств
по учебной дисциплине
«Основы программирования»**

в рамках основной профессиональной образовательной программы (ОПОП)
по программе подготовки квалифицированных рабочих, служащих
по профессии 09.01.03 Мастер по обработке цифровой информации

Вологда,

2017 год

I. Паспорт комплекта контрольно-измерительных средств

1. 1. Область применения комплекта контрольно-измерительных средств

Комплект оценочных средств предназначен для оценки результатов освоения учебной дисциплины «Основы программирования» по профессии 09.01.03 Мастер по обработке цифровой информации

КОС включают контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме экзамена

2. Результаты освоения дисциплины, подлежащие проверке

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
Знание: этапов решения задачи на компьютере; типов данных; базовых конструкций изучаемых языков программирования Умение: работать в среде программирования	Оценка устного и письменного опроса Оценка тестирования Оценка выполнения практических работ Оценка результатов внеаудиторной самостоятельной работы Оценка контрольной работы
Знание: принципов структурного и модульного программирования; принципов объектно-ориентированного программирования Умение: реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	Оценка устного и письменного опроса Оценка тестирования Оценка выполнения практических работ Оценка результатов внеаудиторной самостоятельной работы

.3. Распределение оценивания результатов обучения по видам контроля

Наименование элемента умений или знаний

У 1. Работать в среде программирования

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков

34. Принципы структурного и модульного программирования

35. Принципы объектно-ориентированного программирования

Контрольно-оценочные материалы по учебной дисциплине

«Основы программирования»

Практическая работа

Пояснения к работе: При вводе числа и символы можно набрать как на одной строке, так и на различных строках. При этом ввод со следующей строки осуществляется в том случае, если предыдущей процедурой является READLN.

Переменной логического типа можно присвоить значение либо в разделе констант, либо в операторе присваивания. Вводить значения переменных логического типа нельзя.

Перед процедурами ввода следует использовать процедуру вывода на экран приглашения – подсказки о том, что наступила время ввода информации и какой именно информации.

Например: WRITELN ('ВВЕДИ ЦЕЛЫЕ ЧИСЛА ('K, L, M)').

Выводить данные следует двумя способами: в стандартной форме и с указанием ширины поля вывода, при этом выводимую информацию необходимо снабжать заголовками.

5. Пример программы.

Исходные данные: K=-73, L=12, M=21425, X=-4192.21, Y=21.345, S1='*', S2='V', S3='F'.

```
PROGRAM INOUT;
CONST LOG=TRUE;
VAR K,L,M^ INTEGER; X,Y,: REAL; S1,S2,S3: CHAR;
BEGIN
WRITELN ('ВВЕДИ ЦЕЛЫЕ ЧИСЛА K,L,M');
READ (K,L,M);
WRITELN (' :30, 'СТАНДАРТНЫЙ ФОРМАТ,' :12,'ЗАДАННАЯ ШИРИНА
ПОЛЯ');
WRITELN(' ',K,L,M,' :5,'K=',K:3,'L=',L:2,'M=',M:5); WRITELN;
WRITELN ('ВВЕДИ ВЕЩЕСТВЕННЫЕ ЧИСЛА X,Y'); READLN (X,Y);
WRITELN (' :25,'ВЕЩЕСТВЕННЫЕ ЧИСЛА');
WRITELN (' :5,'СТАНДАРТНЫЙ ФОРМАТ,' :18,'ЗАДАННАЯ ШИРИНА
ПОЛЯ');
WRITELN(' ':5,X,Y,' ':10,'X=',X:8:2,'Y=',Y:6:3); WRITELN;
WRITELN ('ВВЕДИ СИМВОЛЫ S1,S2,S3:'); READLN (S1,S2,S3);
WRITELN ('ПЕЧАТЬ СИМВОЛЬНЫХ ПЕРЕМЕННЫХ С ШИРИНОЙ ПОЛЯ
5:',S1:5,S2:5,S3:5);
WRITELN ('ЛОГИЧЕСКАЯ ПЕРЕМЕННАЯ=' ,LOG);
END.
```

Задание:

Организовать ввод и вывод данных заданных типов.

Время на выполнение: 90 минут

Проверяемые умения:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание: Составить программу алгоритма

Ввести текст программы в окне редактора.

Выполнить компиляцию программы.

Исполнить программу и просмотреть ее выполнение на экране.

Сохранить программу.

Пояснения к работе:

В Паскале имеются две реализации одной из основных алгоритмических структур - ветвления. Это условный оператор и оператор выбора. Условный оператор имеет вид

If Boolean then statement; где boolean представляет собой логическое выражение, которое принимает одно из двух возможных значений – «истина» (True) или «ложь» (False). В роли логического выражения обычно выступает какое-то условие, которое может выполняться (результат – «истина») либо нет (результат – «ложь»). Если логическое выражение Boolean принимает значение «истина», то выполняется оператор statement. В противном случае выполняться будет оператор, следующий за условным оператором.

Следует отметить, что согласно формальным правилам языка в условном операторе после then допускается применение только одного оператора. Если при выполнении условия следует выполнить несколько операторов языка, используется составной оператор.

Операторы if ... then ... можно вкладывать друг в друга. Полная версия условного оператора имеет следующий вид: If Boolean then statement1 else statement2;

Если выражение Boolean принимает значение «истина», то управление передается на оператор statement1, если же нет, то на оператор statement2.

Для ситуаций, где имеется несколько альтернатив, можно использовать оператор case. Этот оператор называется оператором выбора и имеет следующий вид:

```
Case expression of  
List_of_values_1: statement_1;  
List_of_values_2: statement_2;...  
List_of_values_n: statement_n;  
Else statement;  
End;
```

Здесь между зарезервированными словами case и of находится выражение expression, принимающее значение, которое может находиться в одном из списков значений List_of_values..., находящихся слева от двоеточий. Данное выражение называется селектором. Каждый оператор, идущий за двоеточием, отделяется от следующего списка значений точкой с запятой. Ветвь else, отвечающая всем перечисленным значениям выражения expression, необязательна. При выполнении данного оператора вначале вычисляется значение селектора. Затем выбирается тот список, которому принадлежит полученное значение, и выполняется соответствующий оператор. В списках значений оператора case допустимыми являются скалярные типы переменных, включая целые и исключая вещественные типы. Любое заданное значение селектора может входить в список значений неоднократно, но выполняться будет лишь первая подходящая ветвь. Если значение селектора отсутствует в списках значений, ни одна из альтернатив выполняться не будет. В этом случае выполняется ветвь else оператора case или (если эта ветвь отсутствует) следующий за case оператор.

Задание: Составить программу, которая из двух вводимых целых чисел печатает заключение о том, какое число больше.

Составьте программу, проверяющую, верно ли утверждение, что введенное вами целое число является четным.

Составьте программу, проверяющую, верно ли утверждение, что введенное вами целое число делится без остатка на 3.

Напишите программу, которая анализирует человека по возрасту и относит к одной из четырех групп: дошкольник, ученик, работник, пенсионер. Возраст вводится с клавиатуры.

Составьте программу, определяющую, входит ли введенная вами цифра в десятичную запись введенного вами трехзначного числа, и печатающую сообщение о том, входит ли эта цифра в запись числа или нет.

Время на выполнение: 90 минут

Проверяемые знания и умения:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –
За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание:

Составить программу алгоритма (по вариантам).

Ввести текст программы в окне редактора.

Выполнить компиляцию программы.

Исполнить программу и посмотреть ее выполнение на экране.

Сохранить программу.

Пояснения к работе:

Оператор цикла FOR ...DO. Оператор безусловного цикла имеет следующую структуру

```
FOR I:= N1 TO N2 DO
```

```
<Тело цикла>;
```

где I - переменная цикла, которая должна быть перечисляемого типа (целая, символьная и т. д.); N1 - начальное значение переменной цикла;

N2 – конечное значение переменной цикла.

Безусловный цикл выполняется заданное число раз. Чтобы прервать выполнение досрочно, необходимо увеличить I до конечного значения (I:= N2) либо использовать процедуру BREAK, которая прерывает выполнение циклов. Процедура CONTINUE начинает новую итерацию цикла, даже если предыдущая не была завершена.

При программировании может быть использован другой, безусловный оператор цикла, в котором происходит уменьшение значения переменной: FOR I:= N2 DOWNTO N1 DO <тело цикла>;

Оператор цикла REPEAT ... UNTIL. Этот оператор называется циклом с постусловием, т. к. проверка условия осуществляется после каждого выполнения тела цикла:

```
REPEAT <тело цикла> UNTIL <условие>;
```

Цикл выполняется до тех пор, пока условие не станет истинным.

Оператор цикла WHILE ... DO. Этот цикл называется циклом с предусловием, т. к. проверка проводится до начала очередной итерации: WHILE <условие> DO <тело цикла>;

Цикл выполняется, пока условие истинно. Как только условие нарушается, выполнение цикла завершается.

Задания:

Составить программу, которая вычисляет сумму квадратов чисел от 1 до введенного вами целого числа n.

С помощью WHILE напишите программу определения суммы всех нечетных чисел в диапазоне от 1 до 99 включительно.

С помощью цикла while напишите программу определения идеального веса для взрослых людей по формуле: $\text{Ид.вес} = \text{рост} - 100$. Выход из цикла: значение роста=250.

Составьте программу получения в порядке убывания всех делителей данного числа.

Составьте программу определения наибольшего общего делителя двух натуральных чисел.

Напишите программу определения наименьшего общего кратного двух натуральных чисел.

Вычислите сумму квадратов N четных натуральных чисел.

Вводятся по очереди данные о росте N студентов группы. Определить средний рост студентов.

Составить программу вычисления суммы всех двузначных чисел.

Составить программу вычисления факториала натурального числа N.

Составить программу вывода на экран всех простых чисел, не превосходящих заданного N.

Время на выполнение: 90 минут

Проверяемые знания и умения:

У 1. Работать в среде программирования

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание:

Для конкретного варианта составить массив исходных данных.

Выбрав один из алгоритмов упорядочивания, написать программу, которая работает с любым допустимым набором данных.

Входную информацию и результаты счета вывести на экран, снабдив их соответствующими заголовками.

Пояснения к работе:

Массив – упорядоченный набор данных одинакового типа. Массив описывается при помощи задания типа его компонентов и типа индексов. Вещественный тип нельзя использовать при задании типа индексов. Тип компонентов массива может быть любым простым или сложным.

Исходные данные должны включать и положительные числа, и отрицательные, и нули.

Задание:

В массиве из 20 целых чисел найти наибольший элемент и поменять его местами с первым элементом.

В массиве из 10 целых чисел найти наименьший элемент и поменять его местами с последним элементом.

В массиве из 15 вещественных чисел найти наибольший элемент и поменять его местами с последним элементом.

В массиве из 25 вещественных чисел найти наименьший элемент и поменять его местами с первым элементом.

Упорядочить по убыванию массив, содержащий 20 целых чисел.

Упорядочить по невозрастанию массив, содержащий 10 целых чисел.

Упорядочить по убыванию массив, содержащий 15 вещественных чисел.

Упорядочить по невозрастанию массив, содержащий 25 вещественных чисел.

Дан массив целых чисел, содержащий 20 элементов, записать в этот же массив сначала все отрицательные числа и нули, затем все положительные, сохраняя порядок их следования.

Дан массив целых чисел, содержащий 10 элементов, записать в этот же массив сначала все положительные числа, а затем все отрицательные и нули, сохраняя порядок их следования.

Дан массив вещественных чисел, содержащий 15 элементов, записать в этот же массив сначала все отрицательные числа, а затем все положительные и нули, сохраняя порядок их следования.

Контрольные вопросы:

Что такое массив?

Как определить местоположение элемента в массиве?

Что такое индекс? Каким требованиям он должен удовлетворять?

Каким образом задается описание массива, что в нем указывается?

Время на выполнение: 90 минут

Проверяемые умения и знания:

У 1. Работать в среде программирования

-Составление работоспособной программы

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа «Обработка массивов».

Текст практической работы.

Вариант 1.

Что такое массив?

Каким образом задаётся обращение к элементу массива?

Написать фрагмент программы для вычисления

Вариант 2.

Как определить местоположение элемента в массиве?

Что называют инициализацией массива, и зачем она применяется?

Написать фрагмент программы для вычисления

Вариант 3.

Что такое индекс? Каким требованиям он должен удовлетворять?

Каким образом задаётся описание массива, что в нём указывается?

Написать фрагмент программы для вычисления

Вариант 4.

Какие из приведённых описаний одномерных массивов являются неправильными и почему?

```
VAR A: ARRAY [1..20] OF REAL;
```

```
TYPE MAS=ARRAY ['A'..'D'];
```

```
VAR A: MAS;
```

```
VAR A: ARRAY [INTEGER] OF CHAR;
```

Сколько элементов содержит каждый правильно описанный массив в п.1?

Программа. В массиве из 20 целых чисел найти наибольший элемент и поменять его места с первым элементом.

Вариант 5.

Какие из приведённых описаний одномерных массивов являются неправильными и почему?

```
VAR MM: ARRAY [FALSE..TRUE] OF CHAR;
```

```
TYPE SS=ARRAY [-20..0] OF INTEGER;
```

```
VAR MAS: SS;
```

```
TYPE ГОРОД=(КИЕВ, МОСКВА, СОЧИ, АЖЛЕР);
```

```
VAR A: ARRAY [ГОРОД] OF REAL;
```

Сколько элементов содержит каждый правильно описанный массив в п.1?

Программа. В массиве из 10 целых чисел найти наименьший элемент и поменять его места с последним элементом.

Вариант 6.

Какие из приведённых описаний одномерных массивов являются неправильными и почему?

```
VAR A: ARRAY [1..5, 1..7] OF REAL;  
TYPE MAS=ARRAY ['A'..'D', -2..3] OF INTEGER;  
VAR ARR: MAS;  
TYPE MM=ARRAY [1..10] OF CHAR;  
VAR A: ARRAY [2..7] OF MM;
```

Сколько компонентов содержит каждый из описанных выше массивов?

Программа. В массиве из 15 вещественных чисел найти наибольший элемент и поменять его местами с последним элементом.

Вариант 7.

Какие из приведённых описаний одномерных массивов являются неправильными и почему?

```
VAR W: ARRAY [5, 7] OF INTEGER;  
TYPE МАСКА=(ВОЛК, ЛИСА, МЕДВЕДЬ);  
A: ARRAY [МАСКА] OF ARRAY [1..5] OF CHAR;  
VAR MASS: ARRAY[1..5], [1..7] OF REAL;
```

Сколько компонентов содержит каждый из описанных выше массивов?

Программа. В массиве из 25 вещественных чисел найти наименьший элемент и поменять его местами с первым элементом.

Время на выполнение: 90 минут

Проверяемые знания:

31. Этапы решения задачи на компьютере Составление правильного алгоритма.

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание:(выполняется по вариантам)

1. Найти значение функции $Y(X)$ при заданном X . Используя стандартные функции, вычислить $Y11=[Y]$, где $[]$ означают целую часть числа; $Y22=[Y+0.5]$; $Y22=[Y-0.5]$.

2. Записать выражение, зависящее от координат точки $X1$ и $Y1$ и принимающее значение TRUE, если точка принадлежит заштрихованной области, и FALSE, если не принадлежит. Для заданной точки вычислить это выражение и результат выдать на печать.

Пояснения к работе:

Функции, отсутствующие в списке стандартных функций языка Паскаль, следует выразить через имеющиеся.

Печать значения выражения в данной точке организовать, используя запись выражения в операторе WRITELN.

Пример программы, которая вычисляет $Y(X)=3-x+1\sin(X)$ при заданном X , печатает $Y11$, $Y22$ и проверяет принадлежность точки с координатами $(X1, Y1)$ заштрихованной области. Исходные данные: $X=-1.5$, $X1=0.5$, $Y1=1.2$.

```
Y10 1 X PROGRAM FUN;  
VAR X,X1,Y,Y1:REAL; BEGIN WRITELN ('ВВЕДИ X:');  
READLN (X);  
Y:=EXP((-X+1)*LN(3))*SIN(X);
```

```

WRITELN ('ПРИ X= ',X:8:3, 'Y=',Y:8:4);
WRITELN (' Y11=',TRUNC(Y):4,' Y22=',ROUND(Y):4);
WRITELN ('ВВЕДИ КООРДИНАТЫ: X1,Y1');
READLN (X1,Y1);
WRITELN ((X1>=0) AND (Y1>=0) AND (SQR(X)+SQR(Y)<=1));
END.

```

Контрольные вопросы:

Какие стандартные функции определены для целых чисел?

Какие функции преобразуют вещественный аргумент в целое число? Чем они отличаются?

Что такое выражение, операция, операнд?

Какие операции называются операциями отношения? В чем заключаются особенности результата операций отношения?

Каковы основные правила для определения старшинства операций?

Время на выполнение: 90 минут

Проверяемые знания и умения:

У 1. Работать в среде программирования -Составление работоспособной программы

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа по теме «Оператор варианта»

Текст практической работы.

1 вариант

```

type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, декабрь);

```

```

день=1..31;

```

```

var d1,d2: день; m1,m2: месяц; t: Boolean;

```

Переменной t присвоить значение true, если дата d1, m1 предшествует (в рамках года) дате d2, m2, и значение false иначе.

2. Определить, какими будут значения переменных p и d после выполнения операторов:p:=true; d:=1;

```

case k mod 10 of 3, 2, 7, 2: d:=k;

```

```

1;

```

```

4, 8: begin p:=false; d:=2 end;

```

```

9, 6: begin p:=false; d:=3 end

```

```

end;

```

если целая переменная k имеет значение 6.

```

3. type цвет=(чёрный, серый, белый);

```

```

var c: цвет;

```

Напечатать значение переменной c.

2 вариант

```

type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, декабрь);

```

```

день=1..31;

```

```

var m, m1: месяц; n:1..12;

```

Переменной *m1* присвоить название месяца, следующего за месяцем *m* (с учётом того, что за декабрём идёт январь)

2. Определить, какими будут значения переменных *p* и *d* после выполнения операторов:

```
p:=true; d:=1;
case k mod 10 of 3, 2, 7, 2: d:=k;
1: ;
4, 8: begin p:=false; d:=2 end;
9, 6: begin p:=false; d:=3 end
end;
```

если целая переменная *k* имеет значение 235.

3. var *k*: 1..9;

Напечатать значение переменной *k* римскими цифрами.

3 вариант

1. type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь);

день=1..31; var *m*, *m1*: месяц; *n*: 1..12; Переменной *m1* присвоить название *k*-го месяца после месяца *m*

2. Определить, какими будут значения переменных *p* и *d* после выполнения операторов:

```
p:=true; d:=1;
case k mod 10 of 3, 2, 7, 2: d:=k;
1: ;
4, 8: begin p:=false; d:=2 end;
9, 6: begin p:=false; d:=3 end
end;
```

если целая переменная *k* имеет значение 71.

3. type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь);

var *d*: 28..31; *m*: месяц;

Переменной *d* присвоить количество дней в месяце *m* (год считать невисокосным).

4 вариант

1. type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь);

день=1..31;

var *m*, *m1*: месяц; *n*: 1..12;

Переменной *m1* присвоить название *n*-го месяца года.

2. Определить, какими будут значения переменных *p* и *d* после выполнения операторов:

```
p:=true; d:=1;
case k mod 10 of
3, 2, 7, 2: d:=k;
1: ;
4, 8: begin p:=false; d:=2 end;
9, 6: begin p:=false; d:=3 end
end; если целая переменная k имеет значение 100.
```

3. type месяц=(январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь);

var *k*: 1..366; *d*: 28..31; *m*: месяц;

Определить *k*- порядковый номер того дня високосного года, который имеет дату *d*, *m*.

Время на выполнение: 90 минут

Проверяемые знания:

31. Этапы решения задачи на компьютере Составление правильного алгоритма.

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Тест «Использование стандартных функций и процедур для работы со строками»

Каков будет результат выполнения программы:

```
var s1, s2, s3: string;
```

```
begin
```

```
s1:=’паро’; s2:=’воз’;
```

```
s3:=concat(s1, s2);
```

```
writeln(s3);
```

```
end.
```

1) пар и воз 2) парвз 3) 7 4) паровоз

Каков будет результат выполнения программы:

```
var s1, s2: string;
```

```
begin
```

```
s1:=’информатика’; delete(s1, 3, 4);
```

```
writeln(s1); end.
```

инатика 2) форма 3) инф 4) инфо

Каков будет результат выполнения программы:

```
var s1, s2: string;
```

```
begin
```

```
s1:=copy(’крокодил’,4,3); writeln(s1); end.
```

крок 2) одил 3) код 4) кродил

Каков будет результат выполнения программы:

```
var s: string;
```

```
begin s:=length(’каникулы’); end.
```

s=0 2) s=1 3) s=8 4) s=true

Каков будет результат выполнения программы:

```
var r: real;
```

```
begin
```

```
r:=4.869; t:=trunc(r); end.
```

t=23.07 2) t=2.207 3) t=5 4) t=4

Какая из данных программ записана без ошибок:

```
var b:Boolean; begin b:=7; writeln(’результат: ’,
```

```
b); end.
```

```
var b:Boolean; begin b:=false; if not b then writeln(’Ура!’);
```

```
end.
```

```
var b:Boolean; begin b:=’Hello, World’; writeln(b); end.
```

```
var b:Boolean; c: real; begin c:=sqrt(b); writeln(’результат: ’,
```

```
c); end.
```

Какая из данных программ на Паскале правильная:

```
var r:string; begin r:=true; if r then halt; end.
```

```
var r:string; c:char; begin r:=4/c; end.  
var r:string; begin r:='Hello World!'; writeln (r); end.  
var r:string; begin r:=Hi, friend; writeln (r); end.
```

Выберите правильный результат действия программы на Паскале:

```
const n=2;  
var k:integer; m,l: real;  
begin  
l:=0;  
for k:=1 to 6 do  
m:=k/n;  
l:=l+m;  
end.
```

1) l=10.5 2) l=20.5 3) l=1.5 4) l=10

Проверяемые знания:

31. Этапы решения задачи на компьютере Составление правильного алгоритма.

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание:

Составить программу алгоритма (по вариантам).

Ввести текст программы в окне редактора.

Выполнить компиляцию программы.

Исполнить программу и просмотреть ее выполнение на экране.

Сохранить программу на диске.

Пояснения к работе:

Множественный тип данных S строится на основе некоторого базового типа BS, в качестве которого используется любой ограниченный или скалярный тип, кроме неограниченного целого и вещественного типов.

В общем виде множественный тип данных задается следующим образом: TYPE S = SET OF BS. Базовый тип BS определяет конечное множество значений множественного типа S. Любые подмножества базового множества могут являться значениями переменных множественного типа S. Они образуются перечислением через запятую элементов базового множества или выражений базового типа и заключаются в квадратные скобки. Множество, не содержащее элементов, называется пустым и обозначается как [].

Варианты задания:

Дана непустая последовательность символов. Требуется построить и напечатать множество, элементами которого являются встречающиеся в последовательности:

цифры от '0' до '9';

буквы от 'A' до 'F' и от 'X' до 'Z';

буквы от 'G' до 'N' и цифры от '0' до '9';

знаки препинания;

буквы от 'A' до 'Z' и цифры от '0' до '5';

буквы от 'T' до 'X' и знаки препинания;

цифры от '5' до '9' и знаки арифметических операций;

знаки арифметических операций и знаки препинания;

цифры и знаки арифметических операций;

знаки препинания и буквы от 'E' до 'N';

знаки операций отношения.

Контрольные вопросы:

Какие типы данных используются в качестве базовых при построении множественных типов?

Какие операции определены над переменными множественного типа и каков их приоритет?

Время на выполнение: 90 минут

Проверяемые умения и знания:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Пояснения к работе:

Запись — это структура данных, доступ к элементам которой осуществляется по имени (названию элемента). Элементы записи могут иметь разный тип, поэтому при описании записи надо указывать и название каждого элемента, и его тип.

Описание записи имеет следующий вид:

тип название-типа-записи = record

название-элемента: тип-элемента;

end;

В качестве названий элементов выступают обычные идентификаторы Паскаля -так же, как при описании переменных. Например, описание записи, характеризующей футбольную команду (содержащую название команды, города, страны и год создания) может выглядеть так:

type TFootballTeamRecord = record

Name: TFootballTeam;

City, Country: String;

Year: 1800..3000; end;

Элемент записи Year (год) можно описать как имеющий тип integer, но более корректно явно указать его допустимый диапазон значений.

Для обращения к элементу записи сначала указывается имя переменной, затем точка и название соответствующего элемента.

var T: TFootballTeamRecord;

begin

T.Name := Spartak;

T.City := 'Москва';

Когда надо задать значения большому числу элементов переменной-записи, каждый раз предварительно указывать ее имя неудобно. В Паскале имеется ключевое слово with, которое позволяет опускать имя переменной в логическом блоке. При этом перед каждым упоминанием элемента в этом блоке имя переменной будет ставиться автоматически.

with T do begin

```
Name := Spartak; City := 'Москва'; end;
```

Если в логическом блоке, охваченном словом `with`, требуется использовать также переменную, имя которой совпадает с названием одного из элементов записи, перед ней необходимо указать название модуля (или программы), в котором эта переменная описана.

```
var T; TFootballTeamRecord;  
Name: string;  
Begin  
Name := 'Москва';  
with T do  
begin  
Name := Spartak;  
City := Projectl.Name;  
end;
```

ЗАПИСИ С ВАРИАНТАМИ

Записи, представленные выше, имеют строго определенную структуру. В некоторых случаях это резко ограничивает возможности их применения. Поэтому в языке Паскаль имеется возможность задать тип записи, содержащий произвольное число вариантов структуры. Такие записи называются записями с вариантами. Записи с вариантами обеспечивают средства объединения записей, которые похожи, но не идентичны по форме. Они состоят из фиксированной и вариантной частей.

Использование фиксированной части не отличается от описанного ранее. Вариантная часть формируется с помощью оператора `case`. Он задает особое поле записи— поле признака, которое определяет, какой из вариантов в данный момент будет активизирован. Значением признака в каждый текущий момент выполнения программы должна быть одна из расположенных далее констант. Константа, служащая признаком, задает вариант записи и называется константой выбора.

Формат:

```
Type Rec = record case <поле признака>: <имя типа >of<константа выбора1>:(поле,  
...:тип );
```

```
...
```

```
<константа выбораN>:(поле, ...:тип )
```

```
end;
```

Задания:

Во всех вариантах предусмотреть описание вариантной части записи.

Составьте программу, которая описывает массив записей—телефонный справочник— и обеспечивает ввод данных, поиск номера телефона по фамилии, подсчет и вывод списка всех абонентов по критерию «увлечение компьютерными играми». В записи о каждом однокласснике содержатся следующие сведения: фамилия, имя, телефон, хобби.

Составьте программу, которая описывает таблицу химических элементов, отображая следующую информацию: название, символическое обозначение, массу атома, заряд атомного ядра, перечень основных химических свойств. Программа должна выполнять вывод данных о химическом элементе по указанному символическому обозначению, находить элемент с самой большой массой, с самым маленьким зарядом ядра.

Опишите, используя структуру записи, вступительные экзамены, на которых абитуриенты сдавали три экзамена, а для поступления надо было набрать 12 баллов.

Составьте программу, выводющую на экран следующую информацию: а) список абитуриентов, сдавших все три экзамена на 5

б) список абитуриентов, потерпевших неудачу на экзаменах

в) список абитуриентов, зачисленных в институт

Опишите, используя структуру записи, библиотеку.

Запись, описывающая каждую книгу, должна содержать следующие поля: код книги, название книги, раздел, издательство, год издания и место хранения.

Составьте программу, реализующую следующие функции:

- а) нахождение места хранения определенной книги
- б) выборка всех книг одного автора
- в) выборка всех книг, изданных в определенном городе

Опишите, используя структуру записи, видеотеку.

Запись, описывающая каждую кассету, должна содержать следующие поля: название фильма, жанр, исполнитель главной роли, страна, язык, длительность. Составьте программу, реализующую следующие функции:

- а) выбор всех фильмов определенного жанра
- б) выбор всех фильмов с определенным исполнителем главной роли
- в) выбор всех фильмов длительностью менее 100 мин.

Контрольные вопросы:

Почему запись называют комбинированным типом данных?

Чем определяется объем памяти, требуемый для размещения записи?

Что такое составное имя поля записи? Из каких частей оно состоит и как записывается?

Зачем при обращении к полю записи используется предложение with?

Зачем применяются записи с вариантами? Из каких частей состоит запись с вариантами?

Время на выполнение: 90 минут

Проверяемые умения и знания:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Варианты заданий

Вариант Выражение

1 Заданы два вектора $X(N)$, $Y(N)$ и матрица $A(N, N)$. Найти вектор $Z(N)$, где $Z=XA+YA$

2 Дана последовательность чисел $A(N)$. Сформировать вектор $B(M)$ из чисел, принадлежащих отрезку (X, Y) .

3 Заданы три матрицы $A(N_1, N_1)$, $B(N_2, N_2)$, $C(N_3, N_3)$. Найти минимальное из трех чисел X , Y , Z – следы матриц A , B , C соответственно (следом матрицы называется сумма элементов главной диагонали).

4 Заданы две матрицы $A(N_1, N_1)$, $B(N_2, N_2)$. Выяснить и напечатать, сколько из них является симметричными (матрица называется симметричной, если транспонированная матрица равна исходной)

5 Заданы две матрицы $A(N_1, N_1)$, $B(N_2, N_2)$. Построить таблицу функции $Y=CX+D$ при x , меняющемся от 0 до 1 с шагом 0.1, где C , D – следы матриц A , B соответственно (следом матрицы называется сумма элементов главной диагонали).

6 Заданы два вектора $X(N)$, $Y(N)$. Найти сумму двух векторов, где $C=XA$, а $D=YA$, где A – число, введенное с клавиатуры.

7 Дана матрица $A(N, N)$. Сформируйте два одномерных массива. В один перешлите четные, а в другой нечетные элементы матрицы. Выведите на экран все массивы.

8 Заданы три матрицы $A(N_1, N_1)$, $B(N_2, N_2)$, $C(N_3, N_3)$. Упорядочить по убыванию три числа X , Y , Z , являющихся максимальными элементами матриц

9 Заданы два массива $A(N_1)$, $B(N_2)$. Переменной S присвоить значение -1 , 0 или 1 , если максимальный элемент массива A меньше, равен или больше максимального элемента массива B соответственно

10 Вычислите сумму элементов двумерного массива, индексы которых составляют в сумме заданное число K

11 Из элементов массива $A(N)$ сформировать массив B , элементы которого по модулю больше некоторого значения C

12 Определить среднее арифметическое всех элементов массива $A(N)$, удовлетворяющих условию предыдущей задачи

13 Составить программу для подсчета числа нулевых элементов массива $A(N, M)$. Массив заполнять случайными числами.

14 Составить программу для подсчета числа отрицательных элементов массива $A(N, M)$. Массив заполнять с клавиатуры.

15 Составить программу для подсчета числа положительных элементов массива $A(N, M)$. Массив заполнять случайными числами.

16 Определить среднее арифметическое всех элементов массива $A(N)$, удовлетворяющих условию: $A(I, J) > C$

17 Вычислить сумму элементов массива $A(N, M)$, значения которых по модулю лежат в данном диапазоне: $B \ll C$

18 В массиве $A(N, M)$ заменить числом B значения всех элементов, удовлетворяющих условию $A(I, J) > B$

19 Транспонировать матрицу $A(N, N)$. Транспонированием называется замена элементов строк элементами столбцов).

20 Найти сумму минимальных элементов главной и побочной диагонали

21 Сожмите одномерный массив, удалив предшествующие минимальному элементу

22 Вычислить длину вектора $X(N)$. Длина вектора вычисляется по формуле $L_2 = X(1)^2 + \dots + X(N)^2$

23 Найти сумму максимального и минимального элементов массива $A(N, M)$.

24 Найти сумму максимального и минимального элементов массива $A(N, M)$.

25 Определить количество положительных, отрицательных и нулевых элементов массива $A(N, M)$

26 Создайте матрицу из 15 строк и 15 столбцов. Вычислите произведение суммы элементов главной диагонали на сумму элементов L -й строки

27 Создайте одномерный массив, элементами которого являются суммы положительных элементов строк матрицы

28 Найдите сумму элементов столбца и строки массива, на пересечении которых находится максимальный элемент

Время на выполнение: 90 минут

5.11.3. Перечень объектов контроля и оценки

Проверяемые знания и умения:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа «Организация процедур. Использование процедур»

Задание:

Составить программу :

а) упорядочить по неубыванию массивы разной длины (по вариантам) и вывести их на печать;

б) упорядоченные массивы использовать для построения нового массива с минимальной длиной, новый массив образуется путем перемножения соответствующих компонентов упорядоченных массивов.

Ввести текст программы в окне редактора.

Выполнить компиляцию программы.

Исполнить программу и просмотреть ее выполнение на экране.

Сохранить программу

Пояснения к работе:

1. Заголовок процедуры имеет следующий вид: Procedure name_of_procedure (list_of_parameters);Здесь list_of_parameters представляет собой список параметров процедуры. Имя процедуры не является носителем результата, поэтому тип процедуры в заголовке не описывается. Обращение к процедуре производится простым указанием ее имени в нужном месте программы (но не в составе какого-либо выражения) со списком параметров, если таковой имеется.

2. Если в качестве исходной информации в процедуру передается массив, то его следует передавать по ссылке для экономии памяти, так как в этом случае при вызове процедуры не образуется локальный массив.

3. Несмотря на то что обрабатываемые массивы имеют разную длину, они описываются в программе как массивы одного и того же типа, так как при обращении к процедуре типы соответствующих формальных и фактических параметров должны совпадать.

Контрольные вопросы:

В чем различие между стандартными и определенными пользователем подпрограммами?

Опишите последовательность событий при вызове процедуры.

Что называется параметром, и каково его назначение?

Для чего используется исполнение программы в пошаговом режиме с заходом в процедуры, и как это осуществить?

Каковы особенности параметров-процедур?

Варианты задания:

Ввести и обработать два массива, содержащие соответственно 5 и 8 вещественных компонентов;

Ввести и обработать три массива, содержащие соответственно 3, 6 и 8 целых компонентов;

Ввести и обработать четыре массива, содержащие соответственно 4, 6, 3 и 10 целых компонентов;

Ввести и обработать два массива, содержащие соответственно 4 и 6 вещественных компонентов;

Ввести и обработать три массива, содержащие соответственно 5, 10 и 4 целых компонентов;

Ввести и обработать четыре массива, содержащие соответственно 3, 5, 8 и 6 вещественных компонентов;

Ввести и обработать два массива, содержащие соответственно 7 и 10 вещественных компонентов;

Ввести и обработать три массива, содержащие соответственно 5, 4 и 7 целых компонентов;

Ввести и обработать четыре массива, содержащие соответственно 7, 3, 5 и 4 целых компонентов;

Ввести и обработать два массива, содержащие соответственно 6 и 8 вещественных компонентов;

Ввести и обработать три массива, содержащие соответственно 5, 6 и 9 вещественных компонентов.

Время на выполнение: 90 минут

Проверяемые умения и знания:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Практическая работа

Задание:

Составить программу, которая вычисляет сумму значений заданной в варианте функции $F(x, y)$ в пяти различных точках. Вычисление $F(x, y)$ оформить в виде функции.

Ввести текст программы в окне редактора.

Выполнить компиляцию программы.

Исполнить программу и просмотреть ее выполнение на экране.

Сохранить программу на диске.

Контрольные вопросы:

Чем отличаются локальные и глобальные переменные?

Для чего предназначены функции?

Что включает в себя заголовок функции?

Варианты задания:

Вычислить z-сумму значений функции

$Z = f(a, b) + f(a^2, b^2) + f(a^2-1, b) + f(a-b, b) + f(a^2+b^2, b^2-1)$, где

$u^2 + t^2$, если $u > 0, t > 0$;

$u + t^2$, если $u \leq 0, t \leq 0$;

$f(u, t) = u - t$, если $u > 0, t \leq 0$;

$u + t$, если $u \leq 0, t > 0$

а) $a=2.5; b=-7.3$

б) $a=-0.5; b=4.2$

в) $a=-0.2; b=-0.42$

г) $a=23; b=41.2$

Вычислить z-сумму значений функции

$Z = f(\sin a, b) + f(\cos a, b) + f(\sin^2 a, b-1) + f(\sin a - \cos a, b^2-1) + f(\sin^2 a - 1, \cos a + b)$, где $u + \sin t$, если $u > 0$; $f(u, t) = u + t$, если $u \leq 0$

а) $a = \pi / 18; b = -2.1$

б) $a = 2.3; b = 21.3$

в) $a = -\pi / 14; b = -0.2$

г) $a = -\pi / 10; b = 31.2$

Вычислить z-сумму значений функции

$Z = f(\sin x + \cos y, x + y) + f(\sin x, \cos y) + f(x - y, x) + f(\sin^2 x - 2, a) + f(a+3, b+1)$, где $u + t$, если $u > 1$; $f(u, t) = u - t$, если $0 \leq u \leq 1$;

$t - u$, если $u < 0$ а) $x = \pi / 4; y = 0.41; a = 0.1; b = -2.1$

б) $x = 0.32; y = \pi / 10; a = -0.21; b = 4.2$

в) $x = 19.2; y = 0.48; a = -4.3; b = -6.1$

г) $x = 0.2 \pi; y = 2/3; a = 17.1; b = 0.2$

Время на выполнение: 90 минут

Проверяемые умения и знания:

У 1. Работать в среде программирования -Составление работоспособной программы.

У 2. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования

31. Этапы решения задачи на компьютере

32. Типы данных

33. Базовые конструкции изучаемых языков программирования

34. Принципы структурного и модульного программирования

За верное выполнение работы выставляется положительная оценка –

За неверное выполнение работы выставляется отрицательная оценка – 0 баллов.

Вопросы для экзамена по учебной дисциплине «Основы программирования»

Теоретические вопросы

1. Язык Pascal. Особенности языка.
2. Алфавит языка. Элементарные конструкции.
3. Типы данных. Простые типы данных.
4. Переменные и константы.
5. Математические операции. Логические операции.
6. Операции отношения. Приоритет операций.
7. Основные математические функции.
8. Операторы ввода данных.
9. Операторы вывода данных.
10. Структура программы.
11. Оператор условного перехода if (рассмотреть оба случая).
12. Оператор выбора case, оператор безусловного перехода goto.
13. Арифметические циклы (циклы со счетчиком).
14. Итерационные циклы с предусловием.
15. Итерационные циклы с постусловием.
16. Процедуры в Pascal.
17. Функции в Pascal.
18. Структурированные типы данных. Интервалы.
19. Массивы. Ввод и вывод массивов, основные операции.
20. Строки. Процедуры и функции для работы со строками.
21. Файлы. Процедуры и функции для работы с файлами.

Практические задания

Решите предложенную задачу. Реализуйте решение в виде программы на языке программирования Паскаль.

1. Дано число k ($0 < k < 11$) и таблица размера 4×10 заполненная целыми числами. Найти сумму и произведение элементов k -го столбца данной таблицы.
2. Дана таблица размера 5×9 заполненная целыми числами. Найти суммы элементов в каждой четной строке и каждом четном столбце.
3. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное и максимальное значение в каждой строке.
4. Дана таблица размера 5×10 заполненная целыми числами. В каждой строке найти среднее арифметическое всех элементов этой строки.
5. Дана таблица размера 5×10 заполненная целыми числами. Преобразовать эту таблицу, поменяв местами минимальный и максимальный элемент в каждой строке.
6. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное значение среди сумм элементов ее строк и столбцов и номер строки (столбца) с этим минимальным значением.
7. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальное значение среди максимальных элементов каждой строки и столбца.
8. Дана таблица размера 6×10 заполненная целыми числами. Найти количество ее столбцов, все элементы которых различны.
9. Дана таблица размера 4×9 заполненная целыми числами. Вывести номера строк, содержащих одинаковые элементы.
10. Дана квадратная таблица порядка 5 заполненная целыми числами. Найти суммы элементов расположенных на главной и побочной диагонали.
11. Дана квадратная таблица порядка 5 заполненная целыми числами. Заменить нулями элементы матрицы, лежащие ниже главной и выше побочной диагонали.
12. Дана квадратная таблица порядка 5 заполненная целыми числами. Заменить нулями элементы, лежащие одновременно выше главной диагонали (включая эту диагональ) и выше побочной диагонали (также включая эту диагональ).
13. Дана таблица размера 5×10 заполненная целыми числами. Вывести номера строк, элементы которых монотонно возрастают.
14. Дана таблица размера 5×10 заполненная целыми числами. Найти минимальный среди элементов тех строк, которые упорядочены либо по возрастанию. Если такие строки отсутствуют, то вывести 0.
15. Даны два числа k_1 и k_2 и таблица размера 4×10 заполненная целыми числами. Поменять местами строки таблицы с номерами k_1 и k_2 .
16. Дана таблица размера 5×10 заполненная целыми числами. Поменять местами строки, содержащие минимальный и максимальный элементы таблицы.

17. Дана таблица размера 5×10 заполненная целыми числами. Поменять местами столбец с номером 1 и первый из столбцов, содержащих только положительные элементы.
18. Дано число k и таблица размера 4×10 заполненная целыми числами. Удалить строку таблицы с номером k .
19. Дана таблица размера 5×10 заполненная целыми числами. Удалить столбец, содержащий минимальный элемент таблицы.
20. Дана таблица размера 5×10 заполненная целыми числами. Удалить первый столбец, содержащие только положительные элементы.
21. Дано число k и таблица размера 4×9 заполненная целыми числами. Перед строкой таблицы с номером k вставить строку из нулей.
22. Дана таблица размера 4×9 заполненная целыми числами. Продублировать строку таблицы, содержащий ее минимальный элемент.
23. Дана таблица размера 5×9 заполненная целыми числами. Перед столбцом, содержащим минимальный элемент, добавить столбец, состоящий из единиц.
24. Дана таблица размера 5×10 заполненная целыми числами. Вывести номера строк, элементы которых монотонно убывают.