

Государственное бюджетное образовательное учреждение
дополнительного профессионального образования
«НИЖЕГОРОДСКИЙ ИНСТИТУТ РАЗВИТИЯ ОБРАЗОВАНИЯ»

М. Ю. ВТЮРИН, М. И. ГОЛУНОВА,
Э. В. ЕФИМОВА

ПРАКТИКУМ
по ПРОГРАММИРОВАНИЮ
на ЯЗЫКЕ ПАСКАЛЬ



Часть II

Нижний Новгород
Нижегородский институт развития образования
2015

УДК 372.016:004
ББК 74.263.2
В87

Авторы

М. Ю. Втюрин, канд. физ.-мат. наук, доцент кафедры теории и методики обучения информатике ГБОУ ДПО НИРО;

М. И. Голунова, канд. пед. наук, зав. кафедрой теории и методики обучения информатике ГБОУ ДПО НИРО;

Э. В. Ефимова, канд. пед. наук, доцент кафедры теории и методики обучения информатике ГБОУ ДПО НИРО

*Рекомендовано к изданию
научно-методическим экспертным советом
ГБОУ ДПО НИРО*

Втюрин, М. Ю.
В87 Практикум по программированию на языке Паскаль / М. Ю. Втюрин, М. И. Голунова, Э. В. Ефимова. — Н. Новгород : Нижегородский институт развития образования, 2015. — 28 с.

ISBN 978-5-7565-0669-3

Предлагаемое пособие предназначено для освоения основных алгоритмов обработки массивов и содержит примеры их использования, а также упражнения для самостоятельной работы. Алгоритмы представлены в виде фрагментов программ на языке программирования Паскаль.

Издание адресовано слушателям курсов повышения квалификации в Нижегородском институте развития образования и призвано помочь учителям информатики освоить основы алгоритмизации и программирования с помощью языка Паскаль.

**УДК 372.016:004
ББК 74.263.2**

ISBN 978-5-7565-0669-3

© ГБОУ ДПО «Нижегородский институт развития образования», 2015

Предлагаемое пособие предназначено для освоения основных алгоритмов обработки массивов. Изложение материала основано на использовании языка программирования Паскаль.

В первой главе пособия приведено описание массива как структурированного типа данных и приведены программы, реализующие ввод и вывод элементов массива. Главы, начиная со второй, непосредственно посвящены алгоритмам обработки массивов. Рассмотрены следующие алгоритмы: суммирования, организации счетчиков, поиска минимального элемента, сортировки, построения массива без повторов и другие немаловажные алгоритмы.

Изложение материала ведется по принципу — сначала описываются исходные данные, затем указывается требуемый результат выполнения алгоритма, перечисляются (при необходимости) вспомогательные переменные, приводится алгоритм в виде фрагмента программы на языке программирования Паскаль и после этого для усвоения изложенного рассматривается пример решения конкретной задачи с применением соответствующего алгоритма. В конце каждой главы, начиная со второй, приведены упражнения для самостоятельной работы. Приступая к работе, сначала

изучите программы, приведенные в соответствующем разделе, и попробуйте реализовать их самостоятельно на компьютере, отладить и протестировать с различными исходными данными. После этого рекомендуется самостоятельно приступать к решению задач из упражнений по изучаемой теме.

Пособие предназначено, в первую очередь, для работы на практических занятиях, а также для самостоятельной практической работы. Более полную информацию о языке программирования Паскаль, среде программирования и принципах структурного программирования читатель может получить из источников, представленных в разделе «Литература» данного издания.

Описание массивов.

Ввод-вывод элементов массива

Массив — структурированный тип данных, состоящий из фиксированного числа элементов одного типа.

Описание массивов

var <переменная> : **array** [<тип_индекса>] **of** <тип_переменных>;

Пример:

```
var
  x, y: array [1..25] of integer;
  z: array [-2..8] of real.
```

Описание массивов с помощью констант

Пример:

```
const
  n=10; m=7;
var
  a: array [1..n] of byte;
  b: array [1..m] of real.
```

Ввод-вывод элементов массива (ввод с помощью цикла *for*)

```
program n1;
var
  i: integer;
  a: array [1..10] of real;
begin
```

```
{Ввод 10 элементов массива}
for i:=1 to 10 do
  begin
    write ('a(, i, '=');
    readln (a[i]);
  end;
{Вывод 10 элементов массива}
for i:=1 to 10 do
  writeln (a[i]:6:2);
readln;
end.
```

*Ввод-вывод элементов массива
(ввод с помощью цикла while)*

```
program n2;
var
  i: integer;
  a: array[1..10] of real;
begin
  {Ввод 10 элементов массива}
  i:=1;
  while (i<=10) do
    begin
      write ('a(, i, '=');
      readln (a[i]);
      i:=i+1;
    end;
  {Вывод 10 элементов массива}
  for i:=1 to 10 do
    writeln (a[i]:6:2);
  readln;
end.
```

Подсчет сумм и произведений. Организация счетчиков

Алгоритм нахождения суммы элементов массива

Исходные данные:

n — длина исходного массива;
 $a(n)$ — исходный массив чисел.

Требуемый результат:

s — сумма чисел исходного массива.

Фрагмент программы:

```
s:=0;  
for i:=1 to n do  
  s:=s+a[i];
```

Пример: задан массив из 10 целых чисел. Вычислить сумму элементов массива.

```
program n3;  
var  
  i, s: integer;  
  a: array [1..10] of integer;  
begin  
  {Ввод 10 элементов массива}  
  for i:=1 to 10 do  
    begin  
      write ('a(', i, ')=');  
      readln (a[i]);  
    end;  
  {Подсчет суммы элементов массива}  
  s:=0;  
  for i:=1 to 10 do
```

```

s:=s+a[i];
{Вывод результата}
writeln ('Сумма элементов = ', s);
readln;
end.

```

Алгоритм нахождения произведения элементов массива

▣ **Исходные данные:**

n — длина исходного массива;
a(n) — исходный массив чисел.

▣ **Требуемый результат:**

p — произведение чисел исходного массива.

▣ **Фрагмент программы:**

```

p:=1;
for i:=1 to n do
  p:=p*a[i];

```

Пример: задан массив из 10 целых чисел. Вычислить произведение элементов массива.

```

program n4;
var
  i: integer;
  p: longint;
  a: array [1..10] of integer;
begin
  {Ввод 10 элементов массива}
  for i:=1 to 10 do
    begin
      write ('a(', i, ')=');
      readln (a[i]);
    end;
  {Подсчет произведения}
  p:=1;

```



```

for i:=1 to 10 do
  p:=p*a[i];
  {Вывод результата}
  writeln ('Произведение элементов = ', p);
  readln;
end.

```

Подсчет количества элементов массива, удовлетворяющих некоторому условию

⇒ **Исходные данные:**

n — длина исходного массива;
a(n) — исходный массив чисел.

⇒ **Требуемый результат:**

k — количество элементов массива, удовлетворяющих некоторому условию.

⇒ **Фрагмент программы:**

```

k:=0;
for i:=1 to n do
  if (a[i] ?) then {проверка условия, например: a[i]>0}
    k:=k+1;

```

Пример: задан массив из 10 целых чисел. Подсчитать количество положительных элементов.

```

program n5;
var
  i, k: integer;
  a: array [1..10] of integer;
begin
  {Ввод 10 элементов массива}
  for i:=1 to 10 do
    begin
      write ('a(', i, ')=' );
      readln (a[i]);
    end;

```

```

{Подсчет положительных элементов}
k:=0;
for i:=1 to 10 do
  if (a[i]>0) then
    k:=k+1;
{Вывод результата}
writeln ('Количество положительных эле-
ментов = ', k);
readln;
end.

```

Суммирование элементов массива, удовлетворяющих некоторому условию

▣ Исходные данные:

n — длина исходного массива;
a(n) — исходный массив чисел.

▣ Требуемый результат:

s — сумма чисел исходного массива, удовлетворяющих некоторому условию.

▣ Фрагмент программы:

```

s:=0;
for i:=1 to n do
  if (a[i] ?) then {проверка условия, напри-
мер: (a[i] mod 2=0)}
    s:=s+a[i];

```

Пример: задан массив из 10 целых чисел. Вычислить сумму четных элементов массива.

```

program пб;
var
  i, s: integer;
  a: array [1..10] of integer;
begin

```

```

{Ввод 10 элементов массива}
for i:=1 to 10 do
  begin
    write ('a(, i, '=');
    readln (a[i]);
  end;
{Подсчет суммы четных элементов}
s:=0;
for i:=1 to 10 do
  if (a[i] mod 2 =0) then {Проверка на чет-
                           ность}
    s:=s+a[i];
{Вывод результата}
writeln ('Сумма четных элементов = ', s);
readln;
end.

```

Упражнения

1. Найти сумму элементов массива.
2. Найти произведение элементов массива.
3. Школьный класс писал контрольную по информатике. Подсчитать среднюю оценку по контрольной.
4. Школьный класс писал контрольную по информатике. Подсчитать количество пятерок и количество двоек за контрольную.
5. Школьный класс писал контрольную по информатике. Подсчитать количество удовлетворительных и количество неудовлетворительных оценок, полученных за контрольную.
6. Известно количество каждого вида товара и стоимость единицы каждого товара. Определить суммарную стоимость всех товаров.

7. Задан массив целых чисел. Найти сумму четных чисел.
8. Задан массив целых чисел. Найти произведение нечетных чисел.
9. Задан массив целых чисел. Найти сумму элементов массива, кратных трем.
10. Задан массив целых чисел. Подсчитать количество чисел, кратных семи.
11. Задан массив целых чисел. Подсчитать количество трехзначных чисел.
12. Задан массив целых чисел. Подсчитать количество чисел, оканчивающихся на цифру три.
13. Задан массив целых чисел. Подсчитать сумму двузначных чисел, оканчивающихся на цифру девять.
14. Задан массив действительных чисел. Подсчитать количество целых чисел.
15. Задан массив действительных чисел. Подсчитать сумму нецелых чисел.
16. Задан массив действительных чисел. Подсчитать сумму четных чисел.
17. Задан массив действительных чисел. Подсчитать произведение нечетных чисел.

Поиск минимального (максимального) элемента массива

Определение минимального (максимального) элемента и его порядкового номера в массиве (среди всех элементов)

⇒ **Исходные данные:**

n — длина исходного массива;
a(n) — исходный массив чисел.

⇒ **Требуемый результат:**

min — минимальный элемент массива a(n);
nom — номер минимального элемента в массиве a(n).

⇒ **Фрагмент программы:**

```
min:=a[1];  
nom:=1;  
for i:=2 to n do  
  if a[i]<min then  
    begin  
      min:=a[i];  
      nom:=i;  
    end;
```

Пример: задан массив из 10 целых чисел. Найти минимальный элемент массива и его номер.

```
program n7;  
var  
  i, min, nom: integer;  
  a: array [1..10] of integer;  
begin  
  {Ввод 10 элементов массива}  
  for i:=1 to 10 do
```

```

begin
  write ('a( ', i, ')=');
  readln (a[i]);
end;
{Поиск минимального элемента}
min:=a[1];
nom:=1;
for i:=2 to 10 do
  if (a[i]<min) then
    begin
      min:=a[i];
      nom:=i;
    end;
{Вывод результата}
writeln ('Минимальный элемент = ', min,
        ' Его номер ', nom);
readln;
end.

```

Определение минимального (максимального) элемента и его порядкового номера в массиве (с условием отбора)

▣ **Исходные данные:**

n — длина исходного массива;
a(n) — исходный массив чисел.

▣ **Требуемый результат:**

min — минимальный элемент массива a(n);
nom — номер минимального элемента в массиве a(n).

▣ **Фрагмент программы:**

```

min:=MAXIMUM; { MAXIMUM — величина заведомо большая, чем любой из элементов массива }
nom:=0;

```

```

for i:=1 to n do
  if (a[i] ?) then {условие отбора}
    if a[i] <min then
      begin
        min:=a[i];
        nom:=i;
      end;

```

Пример: задан массив из 10 целых чисел. Найти минимальный нечетный элемент массива и его номер.

```

program n8;
var
  i, min, nom: integer;
  a: array[1..10] of integer;
begin
  {Ввод 10 элементов массива}
  for i:=1 to 10 do
    begin
      write ('a(', i, ')=');
      readln (a[i]);
    end;
  {Поиск минимального элемента}
  min:=32767; {для типа integer}
  nom:=0;
  for i:=1 to 10 do
    if (a[i] mod 2<>0) then
      if (a[i]<min) then
        begin
          min:=a[i];
          nom:=i;
        end;
  {Вывод результата}
  if nom<>0 then
    writeln ('Минимальный нечетный элемент =', min, ' Его номер ', nom)

```

```
else  
    writeln ('Таких элементов нет');  
readln;  
end.
```

Упражнения

1. Определить максимальный элемент массива и его номер. Если таких элементов в массиве несколько, указать тот, который в массиве встретится первым.

2. Определить максимальный отрицательный элемент массива и его номер. Если таких элементов в массиве несколько, указать тот, который в массиве встретится последним.

3. Определить минимальный положительный элемент массива и его номер. Если таких элементов в массиве несколько, указать тот, который в массиве встретится первым.

4. Поменять местами максимальный отрицательный и минимальный положительный элементы массива. В заданном массиве один максимальный отрицательный элемент и один минимальный положительный элемент.

5. Найти среднее арифметическое элементов массива, не равных ни наименьшему, ни наибольшему элементам данного массива.

6. Вывести на экран элементы массива целых чисел, имеющих максимальное количество делителей.

7. Вывести на экран элементы массива целых чисел, имеющих максимальную сумму цифр.

8. Задан массив действительных чисел. Среди всех нецелых элементов этого массива найти наименьший и определить его номер. Если таких элементов в массиве несколько, указать тот, который в массиве встретится первым.

| Сортировка массива

Сортировка элементов массива в порядке возрастания (убывания)

⇒ **Исходные данные:**

n — длина исходного массива;

$a(n)$ — исходный массив чисел.

⇒ **Требуемый результат:**

$a(n)$ — отсортированный массив (по возрастанию или убыванию).

⇒ **Рабочие переменные:**

r — переменная для перестановки элементов массива.

⇒ **Фрагмент программы:**

```
for i:=1 to n-1 do
  for j:=i+1 to n do
    if a[i]>a[j] then {по возрастанию}
      begin
        r:=a[i];
        a[i]:=a[j];
        a[j]:=r;
      end;
```

Пример: задан массив из 10 целых чисел. Упорядочить элементы массива по возрастанию.

```
program n9;
var
  i, j, r: integer;
  a: array[1..10] of integer;
begin
```

```

{Ввод 10 элементов массива}
for i:=1 to 10 do
  begin
    write ('a( ', i, ')=');
    readln (a[i]);
  end;
{Сортировка}
for i:=1 to 9 do
  for j:=i+1 to 10 do
    if a[i]>a[j] then
      begin
        r:=a[i];
        a[i]:=a[j];
        a[j]:=r;
      end;
{Печать результата}
writeln ('После сортировки');
for i:=1 to 10 do
  writeln (a[i]);
readln;
end.

```

Упражнения

1. Упорядочить элементы массива в порядке возрастания.
2. Упорядочить элементы массива в порядке убывания.
3. Вывести на экран элементы массива целых чисел в порядке возрастания их числа делителей.
4. Вывести на экран элементы массива целых чисел в порядке убывания их суммы цифр.
5. Дана ведомость зарплаты сотрудников, в которой указаны табельный номер сотрудника и зар-

плата каждого. Вывести на экран список табельных номеров сотрудников в порядке увеличения их заработной платы.

6. Дан список студентов группы с указанием фамилии, роста и веса каждого студента. Вывести список студентов, упорядоченный по убыванию роста, при равном росте — по убыванию веса. Вместо фамилии можно использовать номер студенческого билета.

7. В деканате составлена ведомость, в которой указаны фамилии студентов, название предметов и количество прогулов по каждому предмету. Вывести на экран фамилии студентов в порядке увеличения суммарного числа их прогулов по всем предметам.

8. Дана таблица стран — участниц олимпийских игр с указанием для каждой из них количества завоеванных золотых, серебряных и бронзовых медалей. Упорядочить все страны по убыванию количества золотых медалей. В случае равенства количества золотых медалей — по убыванию количества серебряных медалей. Если и здесь равенство, то по убыванию количества бронзовых медалей.

Построение массива без повторений

Отбор элементов, удовлетворяющих некоторому условию, в новый массив

▣ **Исходные данные:**

n — длина исходного массива;

$a(n)$ — исходный массив чисел.

▣ **Требуемый результат:**

k — количество элементов формируемого массива;

$b(k)$ — новый формируемый массив.

▣ **Фрагмент программы:**

```
k:=0;
```

```
for i:=1 to n do
```

```
  if (a[i]?) then {условие отбора в новый  
                  массив}
```

```
  begin
```

```
    k:=k+1;
```

```
    b[k]:=a[i];
```

```
  end;
```

Пример: задан массив из 10 целых чисел. Построить новый массив, в который поместить все элементы исходного массива, оканчивающиеся на цифру 3.

```
program n10;
```

```
var
```

```
  i, k: integer;
```

```
  a, b: array [1..10] of integer;
```

```
begin
```

```
  {Ввод 10 элементов массива}
```

```
  for i:=1 to 10 do
```

```

begin
  write ('a(, i, '=');
  readln (a[i]);
end;
{Формирование нового массива}
k:=0;
for i:=1 to 10 do
  if (a[i] mod 10=3) then {отбор по усло-
                          вию}

    begin
      k:=k+1;
      b[k]:=a[i];
    end;
  {Вывод результата}
  if k<>0 then
    for i:=1 to k do
      writeln (b[i])
    else
      writeln ('Массив пуст');
  readln;
end.

```

Поиск заданного элемента в массиве

⇒ Исходные данные:

n — длина исходного массива;

a(n) — исходный массив чисел;

b — элемент, который надо найти в массиве a(n).

⇒ Требуемый результат:

nom — номер элемента b в массиве a(n).

⇒ Фрагмент программы:

```

nom:=0;
i:=1;
while (i<=n) and (nom=0) do

```

```
if a[i]=b then
  nom:=i
else
  i:=i+1;
```

Пример: задан массив из 10 целых чисел. Определить, под каким номером в массиве находится заданное число.

```
program n11;
var
  i, b, nom: integer;
  a: array [1..10] of integer;
begin
  {Ввод 10 элементов массива}
  for i:=1 to 10 do
    begin
      write ('a( ', i, ')=');
      readln (a[i]);
    end;
  writeln ('Введите число ');
  readln (b);
  {Поиск в массиве}
  nom:=0;
  i:=1;
  while (i<=10) and (nom=0) do
    if a[i]=b then
      nom:=i
    else
      i:=i+1;
  {Вывод результата}
  if nom<>0 then
    writeln (b, ' в массиве под номером ',
             nom)
  else
    writeln (b, ' в массиве нет');
  readln;
end.
```

*Формирование массива без повторений
из массива, элементы в котором
могут повторяться*

⇒ **Исходные данные:**

n — длина исходного массива;

$a(n)$ — исходный массив чисел (с повторениями).

⇒ **Требуемый результат:**

k — длина нового массива без повторений;

$b(k)$ — новый массив без повторений.

⇒ **Рабочие переменные:** nom — номер элемента из массива $a(n)$ в массиве $b(k)$.

⇒ **Фрагмент программы:**

```
k:=0;
for i:=1 to n do
begin
  nom:=0;
  j:=1;
  while (j<=k) and (nom=0) do
    if a[i]=b[j] then
      nom:=j
    else
      j:=j+1;
  if nom=0 then
    begin
      k:=k+1;
      b[k]:=a[i];
    end;
end;
```

Пример: задан массив из 10 целых чисел с повторениями. Создать массив без повторений.

```
program n12;
var
  i, j, k, nom: integer;
```

```

a, b:array [1..10] of integer;
begin
  {Ввод 10 элементов массива}
  for i:=1 to 10 do
    begin
      write ('a( ', i, ')=');
      readln (a[i]);
    end;
  {Строим массив без повторов}
  k:=0;
  for i:=1 to 10 do
    begin
      nom:=0;
      j:=1;
      while (j<=k) and (nom=0) do
        if a[i]=b[j] then
          nom:=j
        else
          j:=j+1;
        if nom=0 then
          begin
            k:=k+1;
            b[k]:=a[i];
          end;
        end;
      end;
    {Вывод результатов}
    writeln ('Массив без повторов');
    for i:=1 to k do
      writeln(b[i]);
    readln;
  end.

```


Упражнения

1. Определить, содержит ли числовой массив заданное число.

2. Дан числовой массив с повторениями (может содержать одинаковые элементы). Сформировать из него массив без повторений.

3. Имеются два массива элементов без повторений. Сформировать третий массив без повторений из элементов, содержащихся хотя бы в одном из исходных массивов.

4. Имеются два массива элементов без повторений. Сформировать третий массив из элементов первого, которых нет во втором.

5. Имеются два массива элементов без повторений. Сформировать третий массив из элементов, которые есть в обоих массивах.

6. Дан массив целых чисел, который может содержать повторяющиеся элементы. Найти число различных элементов массива.

7. Дан массив целых чисел, который может содержать повторяющиеся элементы. Выяснить, сколько чисел входит в массив по одному разу.

8. Дан массив целых чисел, который может содержать повторяющиеся элементы. Выяснить, сколько чисел входит в массив более чем по одному разу.

9. Дан массив целых чисел, который может содержать повторяющиеся элементы. Выяснить, имеется ли в массиве хотя бы одна пара совпадающих чисел.

10. Дан массив целых чисел, который может содержать повторяющиеся элементы. Найти число, повторяющееся в массиве максимальное количество раз.

11. Дан массив целых чисел, который может содержать повторяющиеся элементы. Найти число вхождений каждого элемента массива.

12. Дан массив целых чисел, который может содержать повторяющиеся элементы. Удалить из массива все отрицательные элементы.

13. Дан массив целых чисел, который может содержать повторяющиеся элементы. Удалить из массива все элементы, значение которых совпадает с максимальным.

14. Дан массив целых чисел, который может содержать повторяющиеся элементы. Расположить все элементы массива в обратном порядке.

15. Найти упорядоченную по возрастанию последовательность подряд расположенных элементов массива максимальной длины.

16. Найти симметричную последовательность подряд расположенных элементов массива максимальной длины.

1. *Алексеев, Е. Р.* Турбо Паскаль 7. 0 / Е. Р. Алексеев, О. В. Чеснокова, В. Н. Павлыш, Л. В. Славинская. — М. : АСТ : НТ Пресс, 2004.

2. *Громницкий, В. С.* Сборник задач по программированию / В. С. Громницкий, В. А. Гусев, Н. А. Паршин. — Н. Новгород : Издательство ННГУ, 2005.

3. *Дагене, В. А.* 100 задач по программированию : книга для учащихся / В. А. Дагене, Г. К. Григас, К. Ф. Аугутис. — М. : Просвещение, 1993.

4. *Кетков, Ю. Л.* Свободное программное обеспечение: FREE PASCAL для студентов и школьников / Ю. Л. Кетков, А. Ю. Кетков. — СПб. : БХВ-Петербург, 2011.

5. *Культин, Н. Б.* Turbo Pascal в задачах и примерах / Н. Б. Культин. — СПб. : БХВ-Петербург, 2007.

6. *Павловская, Т. А.* Паскаль. Программирование на языке высокого уровня : практикум / Т. А. Павловская. — СПб. : Питер, 2006.

7. *Пильщиков, В. Н.* Сборник упражнений по языку Паскаль : учебное пособие для вузов / В. Н. Пильщиков. — М. : Наука, 1989.

8. Информационный портал для разработчиков Free Pascal [Электронный ресурс]. — Режим доступа URL: www.freepascal.ru.

Содержание _____

Введение

▣▣▣ 3

Описание массивов. Ввод—вывод элементов массива

▣▣▣ 5

Подсчет сумм и произведений. Организация счетчиков

▣▣▣ 7

Поиск минимального (максимального) элемента массива

▣▣▣ 13

Сортировка массива

▣▣▣ 17

Построение массива без повторов

▣▣▣ 20

Литература

▣▣▣ 27

Учебное издание

***ВТЮРИН Максим Юрьевич, ГОЛУНОВА Марина Ивановна,
ЕФИМОВА Элеонора Владимировна***

**ПРАКТИКУМ
по программированию на языке Паскаль**



Часть II

Редактор *И. М. Морева*
Компьютерная верстка *Л. И. Половинкиной*

Оригинал-макет подписан в печать 15.12.2015 г.
Формат $60 \times 84 \frac{1}{16}$. Бумага офсетная. Гарнитура «Times».
Печать офсетная. Усл.-печ. л. 1,9. Тираж 100 экз. Заказ 2296.
ГБОУ ДПО «Нижегородский институт развития образования»
603122, Н. Новгород, ул. Ванеева, 203.
www.niro.nnov.ru

Отпечатано в издательском центре учебной
и учебно-методической литературы ГБОУ ДПО НИРО



