**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

**(БГТУ им.В.Г.Шухова)**

Кафедра программного обеспечения вычислительной техники и

автоматизированных систем

Лабораторная работа №0

дисциплина: Дискретная математика

Выполнил: ст. группы ПВ-201

Машуров Дмитрий Русланович

Проверил: Бондаренко Т.В.

Белгород 2020

**Лабораторная работа №0**

**Задание к работе:**

1. Записать условие задачи
2. Разработать и записать наборы тестовых данных
3. Разработать и представить в виде блок-схемы алгоритм решения задачи
4. Написать и отладить программу на языке программирования. Представить код программы
5. Представить результат работы программы на исходных данных из наборов тестовых данных

**Выполнение:**

Задание варианта №1:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

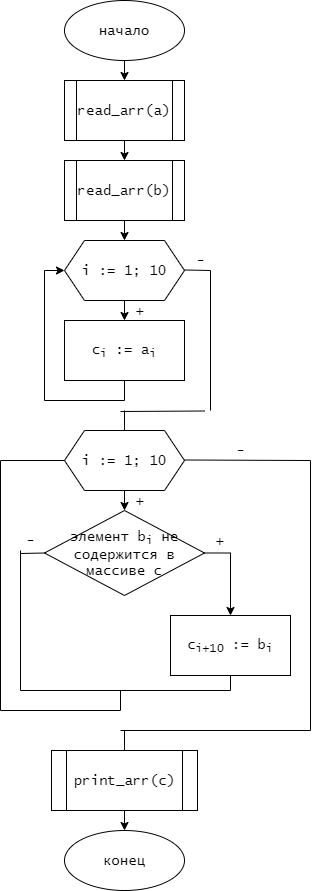
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Получить массив С, содержащий все элементы массивов А и В без повторений.

Наборы тестовых данных:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | C = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14, 15, 16, 18, 19, 20} |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | C = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} |

Алгоритм в виде блок-схемы:



Код программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**procedure** print\_arr(**var** arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**function** is\_elem\_in(arr: t\_out\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

**for** i := 1 **to** 20 **do**

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i, j: integer;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

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

c[i] := a[i];

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

**if not**(is\_elem\_in(c,b[i])) **then**

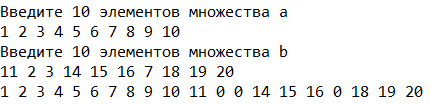
c[i+10] := b[i];

print\_arr(c);

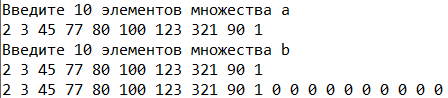
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №2:

А – массив натуральных чисел, в котором нет одинаковых элементов;

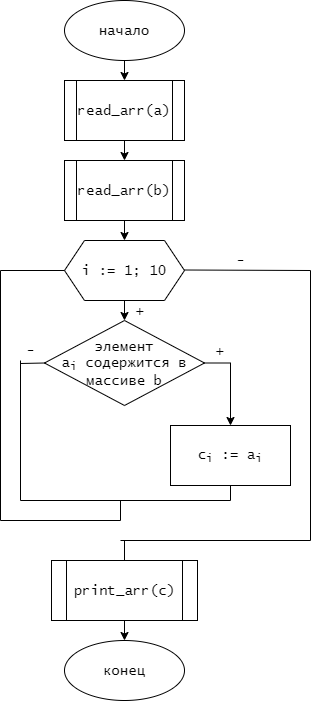
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Получить массив С, содержащий все такие элементы, которые есть и в массиве А и в массиве В.

Наборы тестовых данных:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | C = {2, 3, 7} |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | C = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**procedure** print\_arr(**var** arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i, j: integer;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

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

**if** (is\_elem\_in(b,a[i])) **then**

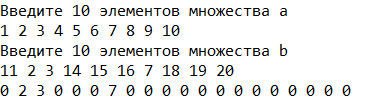
c[i] := a[i];

print\_arr(c);

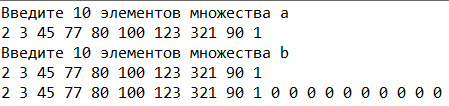
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №3:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

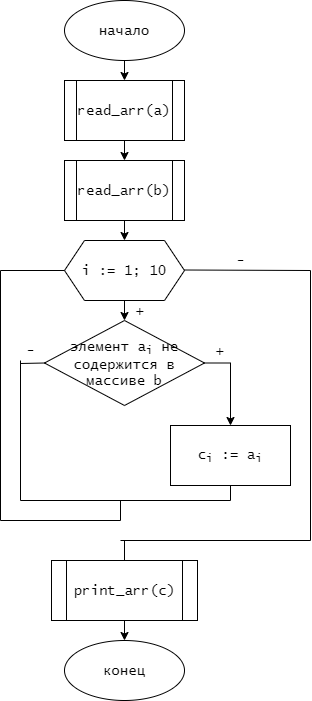
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Получить массив С, содержащий все элементы массива А, которых нет в В.

Наборы тестовых данных:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | C = {1, 4, 5, 6, 8, 9, 10} |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | C = { } |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**procedure** print\_arr(**var** arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i, j: integer;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

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

**if not**(is\_elem\_in(b,a[i])) **then**

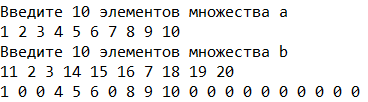
c[i] := a[i];

print\_arr(c);

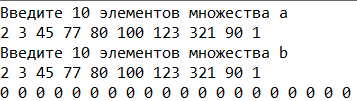
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №4:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

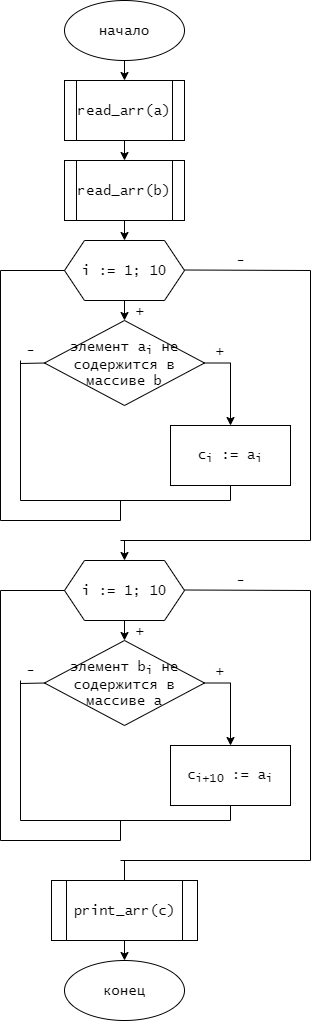
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Получить массив С, содержащий все элементы массива А, которых нет в В и все элементы массива В, которых нет в А.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | C = {1, 4, 5, 6, 8, 9, 10, 11, 14, 15, 16, 18, 19, 20} |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | C = { } |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**procedure** print\_arr(**var** arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i, j: integer;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

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

**if not**(is\_elem\_in(b,a[i])) **then**

c[i] := a[i];

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

**if not**(is\_elem\_in(a,b[i])) **then**

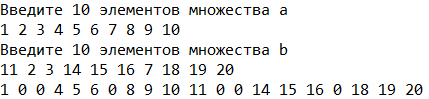
c[i+10] := b[i];

print\_arr(c);

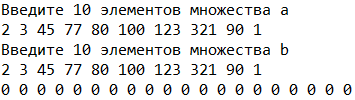
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №5:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

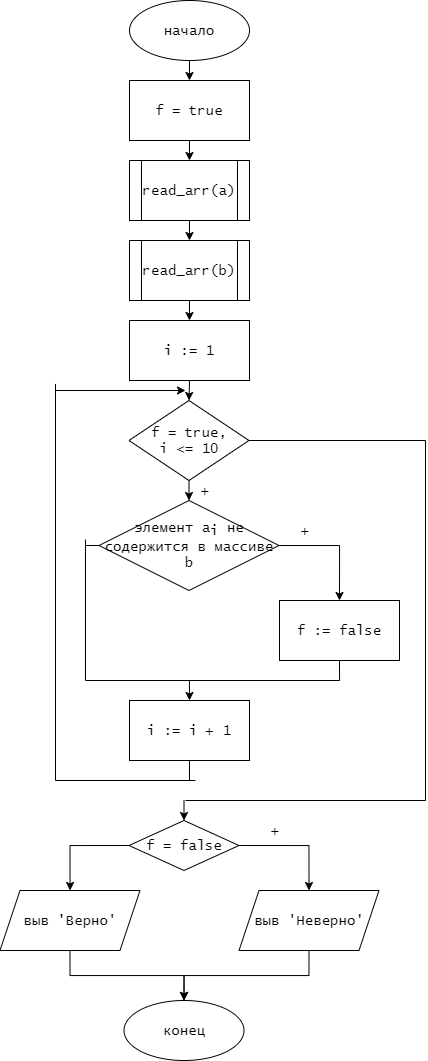
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Определить, верно ли, что массив В содержит каждый элемент массива А.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | Неверно |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | Верно |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

i, j: integer;

f: boolean = true;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

i := 1;

**while** (f = true) **and** (i <= 10) **do**

**begin**

**if not**(is\_elem\_in(b,a[i])) **then**

f := false;

i := i + 1;

**end**;

**if** (f = false) **then**

writeln('Неверно')

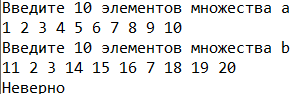
**else**

writeln('Верно');

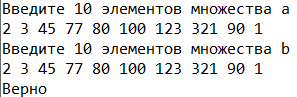
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №6:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

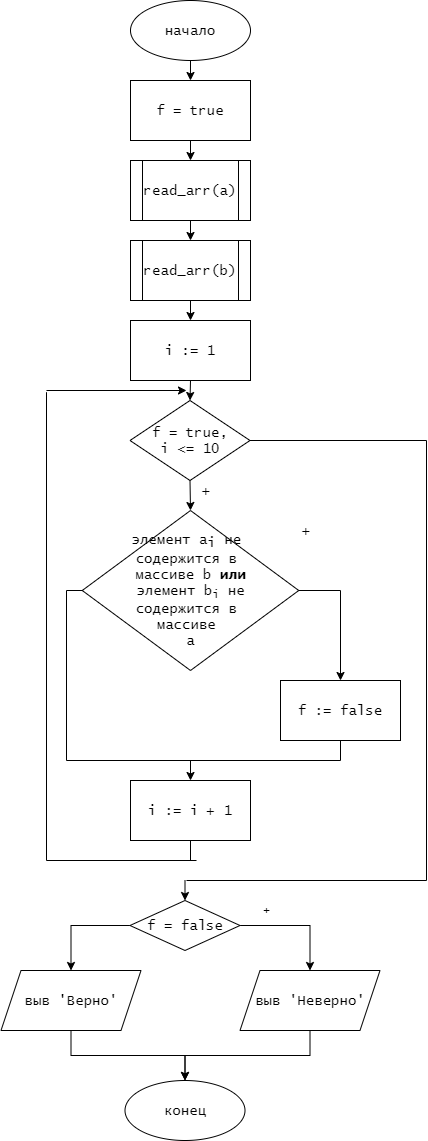
В  – массив натуральных чисел, в котором нет одинаковых элементов.

Определить, верно ли, что массивы А и  В состоят из одинаковых элементов.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | Неверно |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | Верно |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

i, j: integer;

f: boolean = true;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

i := 1;

**while** (f = true) **and** (i <= 10) **do**

**begin**

**if not**(is\_elem\_in(b,a[i])) **or not**(is\_elem\_in(a,b[i])) **then**

f := false;

i := i + 1;

**end**;

**if** (f = false) **then**

writeln('Неверно')

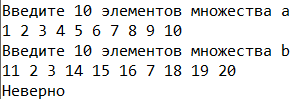
**else**

writeln('Верно');

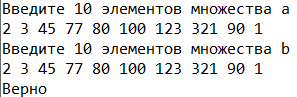
**end**.

Результаты работы программы:

*Пример №1:*



*Пример №2:*



Задание варианта №7:

Дано:

А – массив натуральных чисел, в котором нет одинаковых элементов;

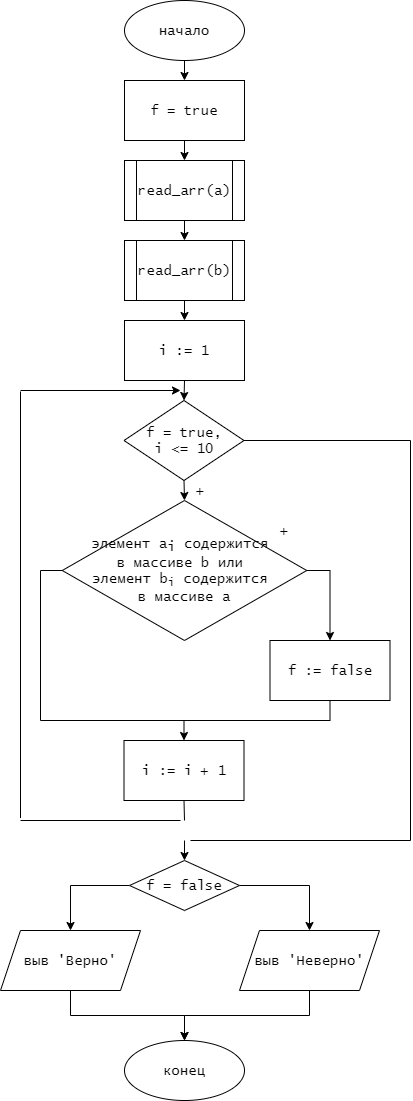
В – массив натуральных чисел, в котором нет одинаковых элементов.

Определить, верно ли, что в массивах А и В нет общих элементов.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | Неверно |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | Неверно |
| 3 | А = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 12, 13, 14, 15, 16, 17, 18, 19, 20} | Верно |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b: t\_in\_arr;

i, j: integer;

f: boolean = true;

**begin**

writeln('Введите 10 элементов множества а');

read\_arr(a);

writeln('Введите 10 элементов множества b');

read\_arr(b);

i := 1;

**while** (f = true) **and** (i <= 10) **do**

**begin**

**if** ((is\_elem\_in(b,a[i])) **or** (is\_elem\_in(a,b[i]))) **then**

f := false;

i := i + 1;

**end**;

**if** (f = false) **then**

writeln('Неверно')

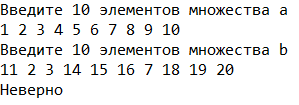
**else**

writeln('Верно');

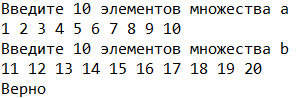
**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*



Задание варианта №8

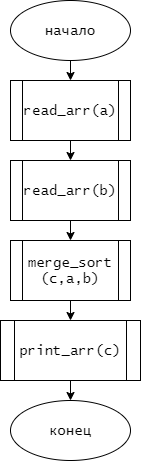
Даны массивы натуральных чисел А и В, упорядоченные по возрастанию.

Получить упорядоченный по возрастанию массив С, содержащий все элементы массивов А и В.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 4, 6, 8, 10, 14, 15, 17, 21, 25}  B = {2, 5, 7, 11, 16, 20, 22, 26, 30, 31} | C = {1, 2, 4, 5, 6, 7, 8, 10, 11, 14, 15, 16, 17, 20, 21, 22, 25, 26, 30, 31} |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**procedure** print\_arr(arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**procedure** merge\_sort(**var** arr: t\_out\_arr; left: t\_in\_arr; right: t\_in\_arr);

**var** i,j: integer;

lp: integer = 1;

rp: integer = 1;

**begin**

j := 1;

**for** i := 1 **to** 20 **do**

**if** (lp <= 10) **and** (left[lp] < right[rp]) **then**

**begin**

arr[j] := left[lp];

lp := lp + 1;

j := j + 1;

**end**

**else if** (rp <= 10) **then**

**begin**

arr[j] := right[rp];

rp := rp + 1;

j := j + 1;

**end**;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

**begin**

writeln('Введите 10 элементов последовательности а');

read\_arr(a);

writeln('Введите 10 элементов последовательности b');

read\_arr(b);

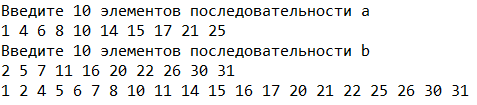
merge\_sort(c,a,b);

print\_arr(c);

**end**.

Результаты работы:

*Пример №1:*



Задание варианта №9:

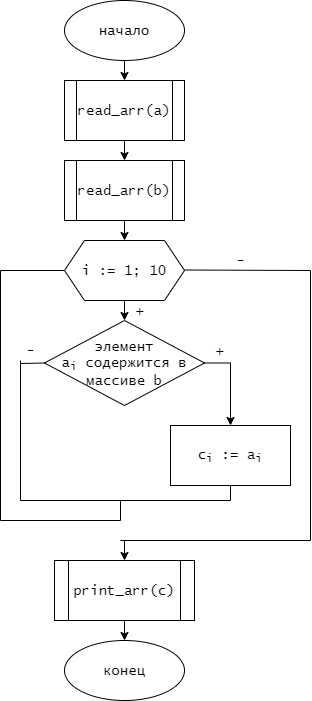
Даны массивы натуральных чисел А и В, упорядоченные по возрастанию.

Получить упорядоченный по возрастанию массив С, содержащий все такие элементы, которые есть и в массиве А и в массиве В.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 5, 6, 11, 16, 20, 22, 27, 30, 35}  B = {2, 5, 7, 11, 16, 20, 22, 26, 30, 31} | C = {5, 11, 16, 20, 22, 30} |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**procedure** print\_arr(arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i: integer;

**begin**

writeln('Введите 10 элементов последовательности а');

read\_arr(a);

writeln('Введите 10 элементов последовательности b');

read\_arr(b);

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

**if** (is\_elem\_in(b,a[i])) **then**

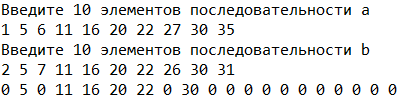
c[i] := a[i];

print\_arr(c);

**end**.

Результаты работы:

*Пример №1:*



Задание варианта №10:

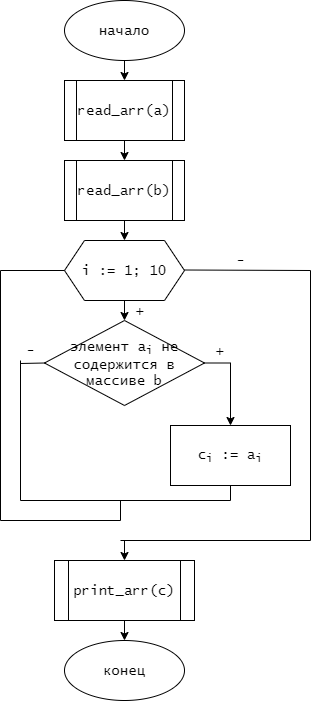
Даны массивы натуральных чисел А и В, упорядоченные по возрастанию.

Получить упорядоченный по возрастанию массив С, содержащий все элементы массива А, которых нет в В.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 5, 6, 11, 16, 20, 22, 27, 30, 35}  B = {2, 5, 7, 11, 16, 20, 22, 26, 30, 31} | C = {1, 6, 27, 35} |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**procedure** print\_arr(arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i: integer;

**begin**

writeln('Введите 10 элементов последовательности а');

read\_arr(a);

writeln('Введите 10 элементов последовательности b');

read\_arr(b);

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

**if not**(is\_elem\_in(b,a[i])) **then**

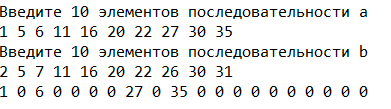
c[i] := a[i];

print\_arr(c);

**end**.

Результаты работы:

*Пример №1:*



Задание варианта №11:

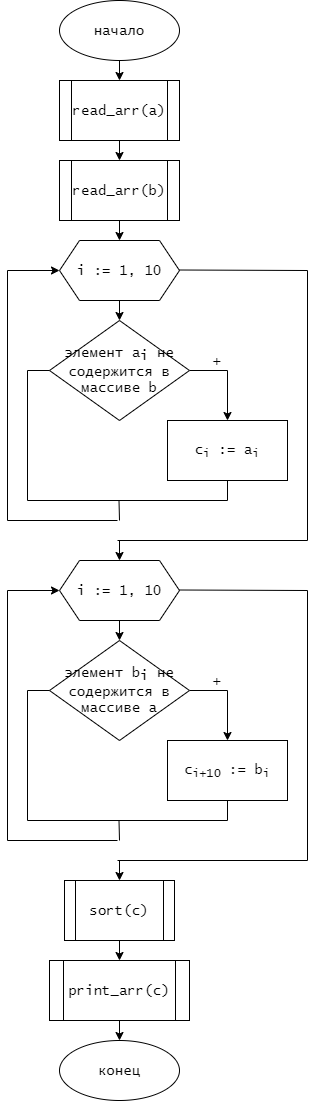
Даны массивы натуральных чисел А и В, упорядоченные по возрастанию.

Получить упорядоченный по возрастанию массив С, содержащий все элементы массива А, которых нет в В и все элементы массива В, которых нет в А.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 5, 6, 11, 16, 20, 22, 27, 30, 35}  B = {2, 5, 7, 11, 16, 20, 22, 26, 30, 31} | C = {1, 2, 6, 7, 27, 31, 35} |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

t\_out\_arr = **array**[1..20] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**procedure** print\_arr(arr: t\_out\_arr);

**var** i: integer;

**begin**

**for** i := 1 **to** 20 **do**

write(arr[i],' ');

**end**;

**procedure** sort(**var** arr: t\_out\_arr);

**var** i,j,key: integer;

**begin**

**for** i := 2 **to** 20 **do**

**begin**

key := arr[i];

j := i;

**while** (j > 1) **and** (arr[j-1] > key) **do**

**begin**

arr[j] := arr[j-1];

j := j - 1;

**end**;

arr[j] := key;

**end**;

**end**;

**var** a,b: t\_in\_arr;

c: t\_out\_arr;

i: integer;

**begin**

writeln('Введите 10 элементов последовательности а');

read\_arr(a);

writeln('Введите 10 элементов последовательности b');

read\_arr(b);

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

**if not**(is\_elem\_in(b,a[i])) **then**

c[i] := a[i];

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

**if not**(is\_elem\_in(a,b[i])) **then**

c[i+10] := b[i];

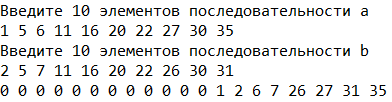
sort(c);

print\_arr(c);

**end**.

Результаты работы:

*Пример №1:*



Задание варианта №12:

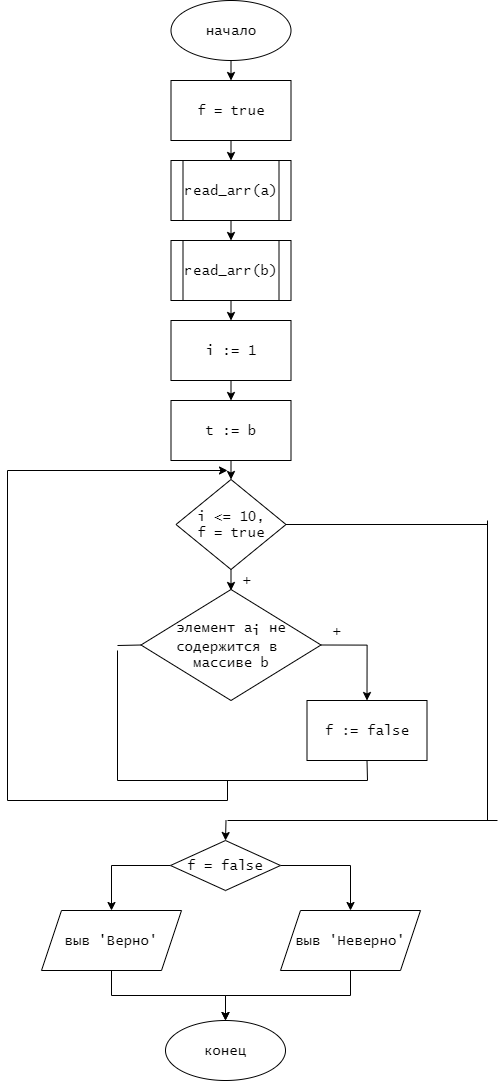
Даны массивы натуральных чисел А и В, упорядоченные по возрастанию.

Определить, верно ли, что массив В содержит каждый элемент массива А.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | A = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 2, 3, 14, 15, 16, 7, 18, 19, 20} | Неверно |
| 2 | A = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1}  B = {2, 3, 45, 77, 80, 100, 123, 321, 90, 1} | Верно |
| 3 | А = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}  B = {11, 12, 13, 14, 15, 16, 17, 18, 19, 20} | Неверно |

Алгоритм в виде блок-схемы:



Текст программы:

**type** t\_in\_arr = **array**[1..10] **of** integer;

**procedure** read\_arr(**var** arr: t\_in\_arr);

**var** i: integer;

**begin**

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

read(arr[i]);

**end**;

**function** is\_elem\_in(arr: t\_in\_arr; elem: integer): boolean;

**var** i: integer;

**begin**

is\_elem\_in := false;

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

**if** (elem = arr[i]) **then**

is\_elem\_in := true;

**end**;

**var** a,b,t: t\_in\_arr;

f: boolean = true;

i,k: integer;

**begin**

writeln('Введите 10 элементов последовательности а');

read\_arr(a);

writeln('Введите 10 элементов последовательности b');

read\_arr(b);

i := 1;

t := b;

**while** (i <= 10) **and** (f = true) **do**

**begin**

**if not**(is\_elem\_in(b,a[i])) **then**

f := false;

i := i + 1;

**end**;

**if** (f = true) **then**

writeln('Верно')

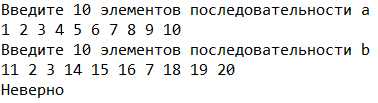
**else**

writeln('Неверно');

**end**.

Результаты работы:

*Пример №1:*



*Пример №2:*

