Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

"Белгородский государственный технологический университет им. В. Г. Шухова"

(БГТУ им. В.Г. Шухова)

Институт энергетики, информационных технологий и управляющих систем

Кафедра программного обеспечения вычислительной техники

и автоматизированных систем

**Лабораторная работа № 0**

**по дисциплине дискретная математика**

**тема: Подготовка к выполнению работы №1.1**

**Выполнил: студент группы ПВ-223**

**Игнатьев Артур Олегович**

**Проверил: доцент   
Рязанов Юрий Дмитриевич**

Белгород 2022

Задание

1. Разработать алгоритм решения задачи (см. варианты ниже).

2. Написать функцию для решения задачи и основную программу.

3. Подобрать тестовые данные. Отладить программу.

Ограничения на алгоритм и программу

1. Глобальные параметры не использовать.

2. Динамическую память не использовать.

3. Все массивы должны быть одного типа. В массивах первые элементы содержат полезную информацию (заданные числа или результат решения задачи), в остальных – мусор. Для каждого массива должна быть переменная, которая хранит количество элементов массива с полезной информацией.

4. В функции не использовать дополнительные массивы и другие структуры данных.

5. Функция не должна вызывать другие функции.

6. Исходные данные (массивы А и В) вводить в основной программе, а не в функции для решения задачи.

7. Результат задачи (массив С) выводить в основной программе, а не в функции для решения задачи.

8. В решении вариантов 8–12 нельзя использовать вложенные циклы, а также цикл, моделирующий вложенный цикл.

Вариант 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.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 3 5 6  5  2 3 5 6 7 |
| Вывод | 1 2 3 4 5 6 | 1 2 3 5 6 |

#include <stdio.h>  
#include <windows.h>  
  
void createArrayСNoRepeatingElements(const long long int \*a, const long long int \*b,  
 long long int \*c, const int n, const int m, int \*h) {  
 for (int i = 0; i < n; i++)  
 \*c++ = a[i];  
 for (int i = 0; i < m; i++) {  
 int x = 1;  
 for (int j = 0; j < n; j++) {  
 if (b[i] == a[j]) {  
 (\*h)--;  
 x = 0;  
 break;  
 }  
 }  
 if (x)  
 \*c++ = b[i];  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int n, m;  
 printf("Введите размер массива a:");  
 scanf("%d", &n);  
  
 long long int a[n];  
 printf("Введите %d значений массива a:\n", n);  
  
 for (int i = 0; i < n; i++)  
 scanf("%lld", &a[i]);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &m);  
  
 long long int b[m];  
 printf("Введите %d значений массива b:\n", m);  
  
 for (int i = 0; i < m; i++)  
 scanf("%lld", &b[i]);  
 int h = n + m;  
  
 long long int c[h];  
 createArrayСNoRepeatingElements(a, b, c, n, m, &h);  
  
 printf("Вывод %d значений массива c:\n", h);  
 for (int i = 0; i < h; i++)  
 printf("%lld ", c[i]);  
 return 0;  
}

Вариант 2

Дано:

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.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 3 5 6  5  2 4 6 7 8 |
| Вывод | 1 2 4 5 | 2 6 |

#include <stdio.h>  
#include <windows.h>  
  
void createArrayСCommonElements(const long long int \*a, const long long int \*b,  
 long long int \*c, const int n, const int m, int \*h) {  
 for (int i = 0; i < m; i++) {  
 for (int j = 0; j < n; j++) {  
 if (b[i] == a[j]) {  
 \*c++ = b[i];  
 (\*h)++;  
 break;  
 }  
 }  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int n, m;  
 printf("Введите размер массива a:");  
 scanf("%d", &n);  
  
 long long int a[n];  
 printf("Введите %d значений массива a:\n", n);  
  
 for (int i = 0; i < n; i++)  
 scanf("%lld", &a[i]);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &m);  
  
 long long int b[m];  
 printf("Введите %d значений массива b:\n", m);  
  
 for (int i = 0; i < m; i++)  
 scanf("%lld", &b[i]);  
 int h = 0;  
  
 long long int c[h];  
 createArrayСCommonElements(a, b, c, n, m, &h);  
  
 printf("Вывод %d значений массива c:\n", h);  
 for (int i = 0; i < h; i++)  
 printf("%lld ", c[i]);  
 return 0;  
}

Вариант 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.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 5 6 7  5  2 3 6 7 8 |
| Вывод | 3 | 1 2 5 6 7 |

#include <stdio.h>  
#include <windows.h>  
  
void createArrayСUniqueElementsA(const long long int \*a, const long long int \*b,  
 long long int \*c, const int n, const int m, int \*h) {  
 for (int i = 0; i < m; i++) {  
 int x = 1;  
 for (int j = 0; j < n; j++) {  
 if (a[i] == b[j]) {  
 x = 0;  
 break;  
 }  
 }  
 if (x) {  
 (\*h)++;  
 \*c++ = a[i];  
 }  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int n, m;  
 printf("Введите размер массива a:");  
 scanf("%d", &n);  
  
 long long int a[n];  
 printf("Введите %d значений массива a:\n", n);  
  
 for (int i = 0; i < n; i++)  
 scanf("%lld", &a[i]);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &m);  
  
 long long int b[m];  
 printf("Введите %d значений массива b:\n", m);  
  
 for (int i = 0; i < m; i++)  
 scanf("%lld", &b[i]);  
 int h = 0;  
  
 long long int c[h];  
 createArrayСUniqueElementsA(a, b, c, n, m, &h);  
  
 printf("Вывод %d значений массива c:\n", h);  
 for (int i = 0; i < h; i++)  
 printf("%lld ", c[i]);  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  3 4 5 6 7 | 5  1 3 4 6 8  5  2 4 5 6 7 |
| Вывод | 1 2 6 7 | 1 3 8 2 5 7 |

#include <stdio.h>  
#include <windows.h>  
  
void arrayNoRepeatingAAndB  
 (const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB,  
 int \*arrayC, int \*const sizeArrayC) {  
  
 int \*cSafeSize = arrayC;  
  
 for (int i = 0; i < sizeArrayA; ++i) {  
  
 int j = 0;  
 while (j < sizeArrayB && arrayA[i] != arrayB[j])  
 j++;  
  
 if (j == sizeArrayB)  
 \*(arrayC++) = arrayA[i];  
 }  
  
 for (int i = 0; i < sizeArrayB; ++i) {  
  
 int j = 0;  
 while (j < sizeArrayA && arrayB[i] != arrayA[j])  
 j++;  
  
 if (j == sizeArrayA)  
 \*(arrayC++) = arrayB[i];  
 }  
  
 \*sizeArrayC = arrayC - cSafeSize;  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB, sizeArrayC;  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 int arrayA[sizeArrayA];  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 int arrayB[sizeArrayB];  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
 int arrayC[sizeArrayC];  
  
 arrayNoRepeatingAAndB(arrayA, sizeArrayA,  
 arrayB, sizeArrayB,  
 arrayC, &sizeArrayC);  
  
 printf("Вывод %d значений массива c:\n", sizeArrayC);  
  
 for (int i = 0; i < sizeArrayC; i++)  
 printf("%d ", arrayC[i]);  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  3 1 5 2 4 | 5  1 2 3 4 5  5  1 2 3 4 6 |
| Вывод | Yes | No |

#include <stdio.h>  
#include <stdbool.h>  
#include <windows.h>  
  
bool AllAToB(const int \*const arrayA, const int arrayASize,  
 const int \*const arrayB, const int arrayBSize) {  
  
 bool conclusion = true;  
  
 for (int i = 0; i < arrayASize && conclusion; i++) {  
 int j = 0;  
 while (j < arrayBSize && arrayA[i] != arrayB[j])  
 j++;  
 conclusion = j != arrayBSize;  
 }  
 return conclusion;  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB;  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 int arrayA[sizeArrayA];  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for(int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 int arrayB[sizeArrayB];  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for(int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
 bool conclusion = AllAToB(arrayA, sizeArrayA,  
 arrayB, sizeArrayB);  
 printf("Содержит ли массив В каждый элемент массива А: ");  
 if (conclusion)  
 printf("YES");  
 else  
 printf("NO");  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  3 1 5 2 4 | 5  1 2 3 4 5  5  1 2 3 4 6 |
| Вывод | Yes | No |

#include <stdio.h>  
#include <stdbool.h>  
#include <windows.h>  
  
bool identicalElements(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB) {  
  
 bool conclusion = sizeArrayA == sizeArrayB;  
  
 for (int i = 0; i < sizeArrayA && conclusion; i++) {  
 int j = 0;  
 while (j < sizeArrayB && arrayA[i] != arrayB[j])  
 j++;  
 conclusion = j != sizeArrayB;  
 }  
 return conclusion;  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB;  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 int arrayA[sizeArrayA];  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 int arrayB[sizeArrayB];  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
 bool conclusion = identicalElements(arrayA, sizeArrayA,  
 arrayB, sizeArrayB);  
 printf("Массивы А и В состоят из одинаковых элементов: ");  
 if (conclusion)  
 printf("YES");  
 else  
 printf("NO");  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  3 1 5 2 4 | 5  1 2 3 4 5  5  6 7 8 9 10 |
| Вывод | No | Yes |

#include <stdio.h>  
#include <stdbool.h>  
#include <windows.h>  
  
bool noCommonElements(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB) {  
  
 bool conclusion = true;  
  
 for (int i = 0; i < sizeArrayA && conclusion; i++) {  
 int j = 0;  
 while (j < sizeArrayB && arrayA[i] != arrayB[j])  
 j++;  
 conclusion = j == sizeArrayB;  
 }  
 return conclusion;  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB;  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 int arrayA[sizeArrayA];  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 int arrayB[sizeArrayB];  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
 bool conclusion = noCommonElements(arrayA, sizeArrayA,  
 arrayB, sizeArrayB);  
 printf("В массивах А и В нет общих элементов: ");  
 if (conclusion)  
 printf("YES");  
 else  
 printf("NO");  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 3 5 7 9  5  2 4 6 8 10 |
| Вывод | 1 2 3 4 5 6 | 1 2 3 4 5 6 7 8 9 10 |

#include <stdio.h>  
#include <windows.h>  
  
#define MAX\_SIZE\_ARRAY 1000  
  
void orderedArrayCElementsArraysAandB(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB,  
 int \*arrayC, int \*sizeArrayC) {  
 int i = 0, j = 0;  
 while (i < sizeArrayA || j < sizeArrayB) {  
 if (j == sizeArrayB || (i < sizeArrayA && arrayA[i] < arrayB[j]))  
 arrayC[(\*sizeArrayC)++] = arrayA[i++];  
 else if (i == sizeArrayA || arrayA[i] > arrayB[j])  
 arrayC[(\*sizeArrayC)++] = arrayB[j++];  
 else  
 i++;  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB, sizeArrayC = 0;  
 int arrayA[MAX\_SIZE\_ARRAY];  
 int arrayB[MAX\_SIZE\_ARRAY];  
 int arrayC[MAX\_SIZE\_ARRAY];  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
  
 orderedArrayCElementsArraysAandB(arrayA, sizeArrayA,  
 arrayB, sizeArrayB,  
 arrayC, &sizeArrayC);  
  
 printf("Вывод %d значений массива c:\n", sizeArrayC);  
 for (int i = 0; i < sizeArrayC; i++)  
 printf("%d ", arrayC[i]);  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 3 6 7  5  3 4 5 6 7 |
| Вывод | 1 2 4 5 | 3 6 7 |

#include <stdio.h>  
#include <windows.h>  
  
#define MAX\_SIZE\_ARRAY 1000  
  
void orderedOnlyAAndB(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB,  
 int \*arrayC, int \*sizeArrayC) {  
  
 int i = 0, j = 0;  
 while (i < sizeArrayA && j < sizeArrayB) {  
 if (i < sizeArrayA && arrayA[i] < arrayB[j])  
 i++;  
 else if (j < sizeArrayB && arrayA[i] > arrayB[j])  
 j++;  
 else  
 arrayC[(\*sizeArrayC)++] = arrayA[i++];  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB, sizeArrayC = 0;  
 int arrayA[MAX\_SIZE\_ARRAY];  
 int arrayB[MAX\_SIZE\_ARRAY];  
 int arrayC[MAX\_SIZE\_ARRAY];  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
  
 orderedOnlyAAndB(arrayA, sizeArrayA,  
 arrayB, sizeArrayB,  
 arrayC, &sizeArrayC);  
  
 printf("Вывод %d значений массива c:\n", sizeArrayC);  
 for (int i = 0; i < sizeArrayC; i++)  
 printf("%d ", arrayC[i]);  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 3 4 6 7  5  2 4 5 6 7 |
| Вывод | 3 | 1 3 |

#include <stdio.h>  
#include <windows.h>  
  
#define MAX\_SIZE\_ARRAY 1000  
  
void orderedArrayCElementsANoB(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB,  
 int \*arrayC, int \*sizeArrayC) {  
  
 int i = 0, j = 0;  
 while (i < sizeArrayA || j < sizeArrayB) {  
 if (j == sizeArrayB || (i < sizeArrayA && arrayA[i] < arrayB[j]))  
 arrayC[(\*sizeArrayC)++] = arrayA[i++];  
 else if (j < sizeArrayB && arrayA[i] > arrayB[j])  
 j++;  
 else  
 i++;  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB, sizeArrayC = 0;  
 int arrayA[MAX\_SIZE\_ARRAY];  
 int arrayB[MAX\_SIZE\_ARRAY];  
 int arrayC[MAX\_SIZE\_ARRAY];  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
  
 orderedArrayCElementsANoB(arrayA, sizeArrayA,  
 arrayB, sizeArrayB,  
 arrayC, &sizeArrayC);  
  
 printf("Вывод %d значений массива c:\n", sizeArrayC);  
 for (int i = 0; i < sizeArrayC; i++)  
 printf("%d ", arrayC[i]);  
  
 return 0;  
}

Вариант 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:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 4 6 7  5  2 3 5 6 7 |
| Вывод | 3 6 | 1 3 4 5 |

#include <stdio.h>  
#include <windows.h>  
  
#define MAX\_SIZE\_ARRAY 1000  
  
void orderedArrayCElementsANoBAndBNoA(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB,  
 int \*arrayC, int \*sizeArrayC) {  
  
 int i = 0, j = 0;  
 while (i < sizeArrayA || j < sizeArrayB) {  
 if (j == sizeArrayB || (i < sizeArrayA && arrayA[i] < arrayB[j]))  
 arrayC[(\*sizeArrayC)++] = arrayA[i++];  
 else if (i == sizeArrayA || arrayA[i] > arrayB[j])  
 arrayC[(\*sizeArrayC)++] = arrayB[j++];  
 else {  
 j++;  
 i++;  
 }  
 }  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB, sizeArrayC = 0;  
 int arrayA[MAX\_SIZE\_ARRAY];  
 int arrayB[MAX\_SIZE\_ARRAY];  
 int arrayC[MAX\_SIZE\_ARRAY];  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
  
 orderedArrayCElementsANoBAndBNoA(arrayA, sizeArrayA,  
 arrayB, sizeArrayB,  
 arrayC, &sizeArrayC);  
  
 printf("Вывод %d значений массива c:\n", sizeArrayC);  
 for (int i = 0; i < sizeArrayC; i++)  
 printf("%d ", arrayC[i]);  
  
 return 0;  
}

Вариант 12

Даны массивы натуральных чисел А и В, упорядоченные по возрастанию. Определить, верно ли, что массив В содержит каждый элемент массива А.

Тестовые данные:

|  |  |  |
| --- | --- | --- |
| Ввод | 5  1 2 3 4 5  5  1 2 4 5 6 | 5  1 2 3 4 5  5  1 2 3 4 5 |
| Вывод | No | Yes |

#include <stdio.h>  
#include <stdbool.h>  
#include <windows.h>  
  
bool arrayBContainsEachElementArrayA(const int \*const arrayA, const int sizeArrayA,  
 const int \*const arrayB, const int sizeArrayB) {  
  
 int i = 0, j = 0;  
 bool conclusion = true;  
  
 while ((i < sizeArrayA || j < sizeArrayB) && conclusion) {  
 if (j == sizeArrayB || (i < sizeArrayA && arrayA[i] < arrayB[j]))  
 conclusion = false;  
 else if (i == sizeArrayA || arrayA[i] > arrayB[j])  
 j++;  
 else  
 i++;  
 }  
 return conclusion;  
}

int main() {  
 SetConsoleOutputCP(CP\_UTF8);  
  
 int sizeArrayA, sizeArrayB;  
  
 printf("Введите размер массива a:");  
 scanf("%d", &sizeArrayA);  
  
 int arrayA[sizeArrayA];  
  
 printf("Введите %d значений массива a:\n", sizeArrayA);  
 for (int i = 0; i < sizeArrayA; i++)  
 scanf("%d", arrayA + i);  
  
 printf("Введите размер массива b:");  
 scanf("%d", &sizeArrayB);  
  
 int arrayB[sizeArrayB];  
  
 printf("Введите %d значений массива b:\n", sizeArrayB);  
 for (int i = 0; i < sizeArrayB; i++)  
 scanf("%d", arrayB + i);  
  
 bool conclusion = arrayBContainsEachElementArrayA(arrayA, sizeArrayA,  
 arrayB, sizeArrayB);  
 printf("Массив В содержит каждый элемент массива А: ");  
 if (conclusion)  
 printf("YES");  
 else  
 printf("NO");  
  
 return 0;  
}