"Ss. Cyril and Methodius" University in Skopje
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Structured programming

Exercises 6

Version 1.0, 25 October, 2016

Table of Contents

1. Arrays []

1.1. Declaring arrays

```
type variable_name[SIZE];

int a[10];
float x[99];
char c[5];
```

1.2. Accessing array element

```
array[element_index];

int a[10];
a[0] = 1; // assigning value 1 of the first element
printf("%d", a[9]); // printing the value of the last element
```

2. Problems

2.1. Problem 1

Write a program that for two arrays read from SI will check if they are equal. Print out the result from the comparison. The maximum size of arrays is 100.

Solution `p6_1_en.c`

```
#include<stdio.h>
#define MAX 100
int main() {
    int n1, n2, element, i;
    int a[MAX], b[MAX];
    printf("First array size:  ");
    scanf("%d", &n1);
    printf("Second array size:  ");
    scanf("%d", &n2);
    if (n1 != n2)
        printf("Arrays are equal\n");
    else {
        printf("Elements of the first array: \n");
        for (i = 0; i < n1; ++i) {
            printf("a[%d] = ", i);
            scanf("%d", &a[i]);
        }
        printf("Elements of the second array: \n");
        for (i = 0; i < n2; ++i) {
            printf("b[%d] = ", i);
            scanf("%d", &b[i]);
        }
        // check if arrays are equal:
        for (i = 0; i < n1; ++i)
            if (a[i] != b[i])
                break;
        if (i == n1)
            printf("Arrays are equal\n");
        else
            printf("Arrays are not equal\n");
    }
    return 0;
}
```

2.2. Problem 2

Write a program that for an array read from SI, will compute the sum of even elements, the sum of odd elements and will compute the ratio even/odd.

Example

For array: `3 2 7 6 2 5 1` The program should print:

```
Sum even: 8
Sum odds: 16
Ratio: 0.75
```

Solution p6_2_en.c

```
#include <stdio.h>
#define MAX 100
int main() {
    int i, n, a[MAX], count_even = 0, count_odd = 0, sum_even = 0, sum_odd = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n; ++i) {
        if (a[i] % 2) {
            count_odd++;
            sum_odd += a[i];
        } else {
            count_even++;
            sum_even += a[i];
        }
    }
    printf("Sum even: %d\nSum odd: %d\n", sum_even, sum_odd);
    printf("Ratio: %.2f\n", (float)count_even / count_odd);
    return 0;
}
```

2.3. Problem 3

Write a program that will compute the scalar product of two vectors with n coordinates. The number of coordinates n and the coordinates are read from SI. Print the result on the SO.

Solution p6_3_en.c

```
#include<stdio.h>
#define MAX 100
int main() {
    int a[MAX], b[MAX], n, i, scalar = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n; ++i)
        scanf("%d", &b[i]);
    for (i = 0; i < n; ++i)
        scalar += a[i] * b[i];
    printf("The scalar product is: %d\n", scalar);
    return 0;
}
```

2.4. Problem 4

Write a program that will check if a given array with n elements read from SI is ascending, descending or neither. Print the result.

Solution `p6_4_en.c`

```
#include <stdio.h>
#define MAX 100
int main() {
    int n, element, a[MAX], i;
    short ascending = 1, descending = 1;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n - 1; ++i) {
        if (a[i] >= a[i + 1]) {
            ascending = 0;
            break;
        }
    }
    for (i = 0; i < n - 1; ++i) {
        if (a[i] <= a[i + 1]) {
            descending = 0;
            break;
        }
    }
    if (!descending && !ascending)
        printf("Array is not ascending and not descending\n");
    else if (descending)
        printf("Array is descending\n");
    else if (ascending)
        printf("Array is ascending\n");
    return 0;
}
```

2.5. Problem 5

Write a program that will remove duplicate from an array. After the transformation print the array.

Solution `p6_5_en.c`

```
#include <stdio.h>
#define MAX 100
int main() {
    int a[MAX], n, i, j, k, removed = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n - removed; ++i)
        for (j = i + 1; j < n - removed; ++j)
            if (a[i] == a[j]) {
                for (k = j; k < n - 1 - removed; ++k)
                    a[k] = a[k + 1];
                removed++;
                --j;
            }
    n -= removed;
    for (i = 0; i < n; ++i)
        printf("%d\t", a[i]);
    return 0;
}
```

3. Source code of the examples and problems

https://github.com/finki-mk/SP/

Source code ZIP