

Самостійна робота 3. Розробка консольних програм, які реалізують обробку масивів

Самостійна робота 3 присвячена поглибленому вивченню матеріалу теми 4 «Масиви». Основна увага при поглибленому вивченні цієї теми приділялась питанню розроблення консольної програми, яка реалізує обробку масиву простих типів даних згідно індивідуального варіанту.

Мета роботи: отримання знань та навичок зі створення в програмному середовищі Visual Studio консольних додатків, які реалізують обробку масивів простих типів даних.

У результаті виконання самостійної роботи у студента формуються такі **компетентності**: знання типових алгоритмів обробки масивів і здатність застосовувати оператори управління мови C # для їх обробки.

Результатом виконання самостійної роботи є налаштований консольний додаток та демонстрація його працездатності.

Завдання для самостійної роботи

Виконувати дану самостійну роботу доцільно перед виконанням лабораторної роботи «Обробка одномірних масивів і матриць».

У ході виконання роботи необхідно:

1. Підготувати чисельні контрольні приклади початкового масиву та результати його обробки згідно з індивідуальним варіантом.
2. Надати словесний опис дії алгоритму та на цій основі розробити його графічну схему (блок-схему).
3. Обґрунтувати вибір операторів мови C#, які найкращим чином відповідають реалізації графічної схеми, та написати відповідний код програми обробки масиву.
4. В середовищі Visual Studio.NET створити шаблон консольного додатку та набрати у вбудований в нього редактор тексту розроблену програму.
5. Відкомпілювати текст програми, усуваючи у разі необхідності помилки.
6. Дослідити роботу програми, аналізуючи виконання контрольних чисельних прикладів.

Контрольні запитання для самодіагностики

1. У чому полягають особливості призначення, оголошення й визначення масиву?
2. Як відтворюється доступ до окремих елементів масиву?
3. Наведіть приклади варіантів ініціалізації масиву.
4. Опишіть загальну схему перебору елементів масиву за допомогою оператора `foreach`.
5. Наведіть приклади алгоритмів пошуку заданих елементів масиву.
6. Наведіть приклади алгоритмів перетворення масиву.
7. У чому суть алгоритму сортування елементів масиву методом «Пузирку»?
8. Як реалізується доступ до елементів двовимірного масиву?
9. У чому суть подання двовимірного масиву як масиву масивів?
10. Наведіть приклади обробки матриць.

Методичні рекомендації до теми

Самостійна робота повинна виконуватися в наступній послідовності.

1. Аналіз індивідуального завдання, підготовка відповідного завданню чисельного початкового прикладу масиву та результатів його обробки.

Наприклад, необхідно написати програму, яка в одновимірному масиві, що складається з 10 елементів типу `double`, обчислює суму елементів, розташованих між першим третім нульовими елементами. У якості початкового масиву потрібно надати масив, який включає в себе три або більш нульових елементів. Також на цьому етапі доцільно визначити ім'я змінних, які надалі будуть використовуватися в програмі, наприклад такі:

```
double m[10 ]; // ім'я поточного масиву  
int cont_zero = 0; // лічильник нульових елементів масиву  
double sum_zeroPrimer zeroTercero = 0; // сума, яку треба обчислити  
int ind_zeroPrimer=0; // індекс першого нульового елементу  
int ind_zeroTercero=0; // індекс третього нульового елементу
```

2. Надати словесний опис дії алгоритму та на цій основі розробити його графічну схему (блок-схему).

При опису алгоритму, необхідно використовуючи ключові фрази типу: «... початковий стан змінних перед початком циклу ...», «... на кожній ітерації проходження масиву виконуються наступні дії ...», «... як результат отримуємо ...».

Графічна схема алгоритму повинна містити виключно стандартні блоки [2] зображення відповідних елементів.

3. Обґрунтувати вибір операторів мови C#, які найкращим чином відповідають реалізації графічної схеми, та написати відповідний код програми обробки масиву.

У якості продовження прикладу, нижче наведено фрагмент програми, який знаходить індекси першого та третього нульових елементів масиву, після чого обчислює суму елементів проміж ними:

```
...
for(int i=0; i<size; i++)
{
    if(m[i]==0)
        ++cont_zero;
    if(cont_zero==1 && ind_zeroPrimer==0)
        ind_zeroPrimer=i;
    if(cont_zero==3 && ind_zeroTercero==0)
        ind_zeroTercero =i;
}
for(int i=ind_zeroPrimer+1; i<ind_zeroTercero; i++)
    sum_zeroPrimer_zeroTercero +=m[i];
...
```

4. В середовище Visual Studio.NET створити шаблон консольного додатку та набрати у вбудований в нього редактор тексту розроблену програму.

5. Відкомпілювати текст програми, усуваючи у разі необхідності помилки.

Особу увагу потрібно надати помилкам, коли значення поточного індексу масиву виходять за межі діапазону.

6. Дослідити роботу програми, аналізуючи виконання контрольних чисельних прикладів.

Література: [1; 3].