

Лабораторна робота № 13

- Тема.** Динамічні масиви
Мета. Формування вмінь і навиків використання динамічних одновимірних масивів. Закріплення вмінь і навиків використання функцій вводу-виводу та обробки одновимірних масивів. Застосування вмінь і навиків програмування алгоритмів лінійної, розгалуженої та циклічної структури.

Контрольні запитання.

1. Коли в програмах використовуються динамічні масиви? Чим вони відрізняються від статичних?
2. Як перевизначити розмір динамічного масиву в програмі? Який тип елементів використовується для перевизначеного масиву?
3. Скільки разів може змінюватися розмір динамічного масиву? Чи можливо змінити тип елемента масиву при перевизначенні?
4. Які значення елементів встановлюються для перевизначеного масиву?
5. Як звільнити пам'ять, що використовується динамічним масивом?

Завдання.

1. Скласти програму формування та друку масиву з N дійсних елементів проміжку $[a,b]$ згідно першого завдання попередньої лабораторної роботи. Після друку масиву збільшити його розмір на чотири елементи і надрукувати ще раз.

Вимоги до програма:

- числові значення N , a , та b ввести з клавіатури з відповідною перевіркою коректності типів та даних;
- масив визначити лише після введення кількості його елементів;
- масив формувати за допомогою оператора присвоєння з використанням датчик випадкових чисел;
- результати формування вивести в одному діалоговому вікні, вказавши для кожного елемента масиву його індекс та значення.

Наприклад, програма для розв'язання завдання варіанту № 2 може мати такий вигляд:

```
int i;
double a, b;
string s="";
a = 4; b = 18;
int N=10;
double[] mas;
povtor:
if (!inputInt(ref N, "Введіть кількість елементів масиву"))
    return;
mas=new double[N];
if (!inputDouble(ref a, "Введіть мінімальне значення"))
    return;
if (!inputDouble(ref b, "Введіть максимальне значення"))
    return;
Random rnd = new Random();
for(i=0; i<N; i++)
    mas[i]=a+rnd.NextDouble()*(b-a);
s = "Масив до перевизначення:\n";
for (i = 0; i < N; i++)
    s += "mas[" + i.ToString() + "]= " + mas[i].ToString() + "\n";
Array.Resize(ref mas, mas.Length +4);
s+= "\nМасив після перевизначення:\n";
for (i = 0; i < mas.Length; i++)
    s+="mas["+i.ToString()+"]=" +mas[i].ToString()+ "\n";
if (MessageBox.Show(s + "\nГенерувати ще?", "Результати",
    MessageBoxButtons.YesNo, MessageBoxIcon.Information)== DialogResult.Yes)
    goto povtor;
```

2. Скласти програму для розв'язання поставленого завдання.

Вимоги до програми:

- числові значення розмірності N , цілочисельних елементів масиву та параметрів обчислень ввести з клавіатури з відповідною перевіркою коректності типів та даних;
- масив визначити лише після введення кількості його елементів;
- для обробки елементів масиву використати оператори циклу;
- результати обчислень вивести в одному діалоговому вікні у вигляді стовпця з елементів вхідного та створених масивів.

Варіанти:

1. Знайти суму елементів одновимірного масиву. Для формування вихідного масиву домножити кожен парний елемент вхідного масиву на отримане значення;
2. Обчислити суми і різниці елементів двох заданих масивів з непарними індексами. Результат надрукувати у вигляді паралельних стовпців;

3. Задано масив X . Обчислити значення функції $y=0,5\ln x$ при непарних значеннях аргумента, заданих в масиві X , і помістити їх в масив Y . Надрукувати елементи масивів X та Y у вигляді паралельних стовпців;
4. Знайти середнє значення елементів заданого масиву. Перетворити вхідний масив, віднімаючи від кожного парного елемента отримане середнє значення;
5. Визначити максимальне значення елементів масиву. Перетворити вхідний масив, віднімаючи від кожного непарного елемента отримане максимальне значення;
6. Сформувати два масиви, включаючи в перший елементи вхідного масиву з парними індексами, а в другий - з непарними;
7. Ввести непарні числа масиву X у зворотній послідовності;
8. Потроїти в масиві всі парні елементи;
9. Визначити елементи масиву, що є подвоєними непарними числами;
10. Визначити елементи масиву, що діляться на 5 і не діляться на 7;
11. Зберегти у вихідному масиві всі непарні елементи;
12. Визначити індекс елемента масиву рівного A . Розділити парні елементи вхідного масиву на цей індекс, а при його відсутності – на 5;
13. Перевірити істинність твердження про те, що кількість від'ємних елементів масиву перевищує кількість додатних. Вивести додатні і від'ємні елементи у вихідні масиви;
14. Знайти квадрати парних елементів масиву;
15. Вхідний масив містить лише значення 0 та 1. Визначити кількість нулів і одиниць в масиві. Поміняти 0 на 1 і навпаки, не застосовуючи оператор розгалуження. Записати у вихідний масив позиції значень 0.

Наприклад, програма для розв'язання завдання варіанту № 1 може бути такою:

```
int i, n=10;
if (!inputInt(ref n, "Введіть кількість елементів масиву"))
    return;
int[] mas = new int[n], output=new int[n];
for (i = 0; i < n; i++)
    mas[i] = i; //ініціалізація масиву
for (i = 0; i < n; i++)
    if (!inputInt(ref mas[i], "Введіть " + i.ToString() + "-й елемент масиву"))
        return;
int suma = 0;
for (i = 0; i < n; i++)
    suma += mas[i];
for (i = 0; i < n; i++)
    if (mas[i] % 2 == 0)
        output[i] = mas[i] * suma;
    else
        output[i] = mas[i];
String rez="Вхідний масив:\n";
for (i = 0; i < n; i++)
    rez += "mas[" + i.ToString() + "]= " + mas[i].ToString() + "\n";
rez += "\nВихідний масив:\n";
for (i = 0; i < mas.Length; i++)
    rez += "output[" + i.ToString() + "]= " + output[i].ToString() + "\n";
MessageBox.Show(rez, "Результати", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

3. Реалізувати алгоритм швидкого пошуку хоча б однієї позиції входження елемента у відсортований масив (додатково 1 бал до рейтингу) або пошуку всіх позицій входження елемента у відсортований масив (додатково 2 бали до рейтингу).

Вимоги до програми:

- числові значення розмірності N , дійсних елементів масиву та елемента для пошуку ввести з клавіатури з відповідною перевіркою коректності типів та даних;
- масив визначити лише після введення кількості його елементів;
- для обробки елементів масиву використати оператори циклу;
- результати обчислень вивести одним оператором.

При цьому фрагмент коду для **бінарного** пошуку позиції входження з консолі може бути, наприклад, такий (введення масиву та перевірку коректності введених даних забезпечте самостійно):

```

//реалізація бінарного пошуку
Console.WriteLine("Введіть елемент для пошуку");
element=Convert.ToDouble(Console.ReadLine());
Array.Sort(mas);
i = 0; j = n - 1; //початкові межі пошуку
while (i<j)
{avg = (i + j) / 2;
  if (mas[avg] == element)
  {i=avg;
   break;
  }
  if (mas[avg] > element)
  j = avg - 1;
  else
  i = avg + 1;
}
Console.WriteLine("Елемент '" + element.ToString() + "' " +
(mas[i]!=element ? "в масиві відсутній" :
"міститься в масиві в позиції " + i.ToString()));

```

Фрагмент коду для **бінарного** пошуку позиції входження та визначення інтервалу входження з діалогових вікон може бути, наприклад, такий (введення масиву та перевірку коректності введених даних забезпечте самостійно):

```

Array.Sort(IS);
double element=IS[0];
element = Convert.ToDouble(Interaction.InputBox("Введіть елемент для пошуку:",
"Введення", IS[0].ToString()));

int i = 0, j = IS.Length - 1, avg;
do
{  avg = (i + j) / 2;
  if (IS[avg] == element)
  {  j = avg;
   break;
  }
  if (element < IS[avg])
  j = avg-1;
  else
  i = avg+1;
} while (i < j);
string res = "", pos="";
if (IS[j] != element)
  pos = "в масиві відсутній";
else
{i = j;
 //пересуваємо i вліво
while (i > 0 && IS[i - 1] == element)
  i--;
 //пересуваємо j вправо
while (j < IS.Length - 1 && IS[j + 1] == element)
  j++;
if (i==j)
  pos="міститься в елементі масиву з індексом "+i.ToString();
else
  pos="міститься в елементах масиву з індексами від "+i.ToString() + " до "+j.ToString();
}
for (i = 0; i < IS.Length; i++)
  res += "IS[" + i.ToString() + "]=" + IS[i].ToString() + "\n";
res+="\nЕлемент '"+element.ToString()+"' " + pos;
MessageBox.Show("Масив:\n" + res, "Результати", MessageBoxButtons.OK, MessageBoxIcon.Information);

```