

Лабораторна робота №5.

Тема: "Функції і масиви"

Мета: Організувати обробку масивів з використанням функцій, навчитися передавати масиви як параметри функцій.

1. Короткі теоретичні відомості

1.1. Функції

Функцію в С можна розглядати:

- як один з похідних типів даних (поряд з масивами й вказівниками);
- як мінімальний виконавчий модуль програми (підпрограму).

Всі функції мають єдиний формат визначення:

<тип><ім'я_функції>(<список_формальних_параметрів>) , де

<тіло_функції> , де

<тип> - або void, якщо функція не повертає значення, або тип значення, що повертається функцією,;

<ім'я_функції> - або main для головної функції, або довільний ідентифікатор, що не збігається зі службовими словами й іменами інших об'єктів програми;

<список_формальних_параметрів> - або порожній (), або список, кожен елемент якого має вигляд:

<позначення_типу><ім'я_параметра>

Наприклад:

(int k)

(char i, char j, int z)

<тіло_функції> - це частина визначення функції, взята у фігурні дужки { }.

Тіло функції може бути або складеним оператором, або блоком. Визначення функцій не можуть бути вкладеними.

Для передачі результату з функції у викликаючу функцію використовується оператор return. Він може використовуватися у двох формах:

- 1) return; - завершує функцію, яка не повертає жодного значення (тобто перед ім'ям функції зазначений тип void)

2) return <вираз>; - повертає значення виразу, вираз повинне мати тип, зазначений перед ім'ям функції.

Якщо програміст не пише оператор return явно, то компілятор автоматично дописує return у кінець тіла функції перед закриваючою фігурною дужкою “}”.

Приклад:

```
int op (char c, int x, int y)
{
switch c
{
case '+': return x+y;
case '-': return x-y;
case '*': return x*y;
case '/': return x/y;
default: cout<<“\ноперация не визначена”;return 0;
}
}
```

Виклик функції здійснюється в такий спосіб:

<позначення функції>(<список фактичних параметрів>); де

<позначення функції> - або ім'я функції, або вказівник на функцію;

<список фактичних параметрів> - список виразів, кількість яких дорівнює числу формальних параметрів функції. Між формальними й фактичними параметрами повинна бути відповідність по типах.

Наприклад:

```
c = op ( '+', 5 ,4 );
```

Синтаксис C передбачає тільки один спосіб передачі параметрів - передача за значенням (тобто змінити значення параметрів всередині функції не можна). Але існує можливість опосередковано змінити значення змінних переданих у вигляді параметрів: за допомогою вказівника у функцію, яка викликається можна передати адресу будь-якого об'єкта із викликаючої програми. Якщо вказівник розіменувати, то отримаємо значення, записане за цією адресою.

Приклад:

1)

//опис функції для обміну змінних a й b

```
void change (int a,int b)
```

```
{
```

```
int r;
```

```
r = a; a = b; b = r;
```

```
}
```

// виклик функції

```
change(a, b);
```

Обміну не відбудеться, тому що результат не буде переданий у викликаючу програму.

2)

```
void change (int *a,int *b)
```

```
{
```

```
int r;
```

```
r = *a; *a = *b; *b = r;
```

```
}
```

// виклик функції

```
change(&a, &b);
```

При виклику передаються адреси, за якими перебувають значення й виконується обмін значень, які перебувають за цими адресами.

1.2. Масиви й рядки як параметри функцій

Якщо в ролі параметр функції використовується позначення масиву, то насправді у функцію передається адреса першого елемента масиву.

Приклад:

//обчислення суми елементів масиву

//варіант 1

```
int sum (int n, int a[] )
```

```
{
```

```

int i,int s=0;
for( i=0; i<n; i++ )
s+=a[i]
return s;
}
void main()
{
int a[]={ 3, 5, 7, 9, 11, 13, 15 };
int s = sum( 7, a );
cout<<s;
}
//варіант 2
int sum (int n, int *a)
{
for(int i=0, s=0; i<n; s+=*(a+i),i++ );
return s;
}
void main()
{
int a[]={ 3, 5, 7, 9, 11, 13, 15 };
int s = sum( 7, a );
cout<<s;
}

```

Рядки в ролі фактичних параметрів можуть бути визначені або як одновимірні масиви типу `char[]`, або як вказівники типу `char*`. На відміну від звичайних масивів у цьому випадку немає необхідності явно вказувати довжину рядка.

2. Постановка завдання

Використовуючи функції, розв'язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр.

3. Варіанти

1. У двовимірному масиві записані слова, що представляють собою послідовність цифр, що завершуються 0. Необхідно роздрукувати слова через кому, взявши надрукований рядок у дужки. Довжина друкованого рядка 60 символів. Добування слова оформити у вигляді функції.

Наприклад:

вихідні дані - 123023402303450

234450234567010

234455677670450

результат - (123,234,23,345)(23445,234567,1)(23445567767,45)

2. Написати функцію для обміну рядків двовимірному масиву з її допомогою відсортувати масив по елементах третього стовпця.
3. Написати процедуру для підсумовування матриць. З її допомогою скласти вихідну матрицю й транспоновану (тобто отриману поворотом вихідної на 90°).
4. Визначити чи є матриця ортонормованою, тобто такою, що скалярний добуток кожної пари різних рядків дорівнює 0, а скалярний добуток рядка самого на себе дорівнює 1.
5. Написати функцію для знищення рядка із двовимірному масиву. Рядки, що залишились, повинні бути розташовані щільно, елементи яких бракує замінюються 0. За допомогою розроблених функцій знищити з масиву рядки з номерами від A до B.
6. Елемент матриці є сідловою точкою, якщо він є найменшим у своєму рядку й найбільшим у своєму стовпці (або навпаки: найбільшим у своєму рядку й найменшим у своєму стовпці). Для заданої матриці визначити всі сідлові точки.
7. Написати процедуру обміну стовпця й рядка двовимірному масиву. З її допомогою поміняти місцями ті рядки й стовпці, перші елементи яких збігаються.
8. Написати функцію транспонування квадратної матриці (тобто повороту початкової матриці на 90°). З її допомогою визначити чи є задана матриця

симетричною. (Матриця називається симетричною, якщо транспонована матриця дорівнює початковій).

9. Написати функцію для обчислення суми елементів квадратної матриці, які розташовані нижче головної діагоналі. З її допомогою знайти максимальне значення такої суми в n матрицях.
10. Написати функцію, що перевіряє чи є від'ємні елементи в зазначеному рядку двовимірному масиву. Знищити з масиву всі рядки з від'ємними елементами, знищений рядок заповнюється 0 і переноситься в кінець масиву.
11. Написати функцію, яка перевіряє чи по зростанню або спаданню впорядкований зазначений рядок двовимірному масиву. Впорядкувати по зростанню всі рядки двовимірному масиву, які не впорядковані по спаданню.
12. Написати функцію, для пошуку максимального елемента в зазначеному рядку двовимірному масиву. Зсунути у двовимірному масиві всі рядки циклічно вправо на кількість елементів, яка дорівнює максимальному елементу в цьому рядку.
13. Визначити чи можна у двовимірному масиві знайти такий стовпець, що розбиває масив на два так, що сума елементів у першому більша, ніж сума елементів у другому. Сам стовпець у розбиті частини не входить.
14. Обчислити добуток всіх стовпців масиву, у яких перший елемент більший від елементів розташованих на головній і бічній діагоналі.
15. Задано двовимірний масив. Знайти суму елементів першого стовпця без одного останнього елемента, суму елементів другого стовпця без двох останніх, суму елементів третього стовпця без трьох останніх і т.д. Останній стовпець не обробляється. Серед знайдених сум знайти максимальну.
16. Задано двовимірний масив $N \times N$. Дозволяється довільно переставляти елементи усередині будь-якого стовпця. Перевірити, чи можна виконавши скінчену кількість перестановок у стовпцях, розташувати на бічній діагоналі елементи так, щоб вони зростали.
17. Задано двовимірний масив $N \times M$. Знайти в ньому підмасив 3×3 , сума елементів якого максимальна. N й M можуть бути не кратні трьом.

18. Задано двовимірний масив $N \times N$. Послідовно розглядаються квадратні підмасиви, правий верхній елемент яких лежить на бічній діагоналі. У кожному такому підмасиві перебуває максимальний елемент. Шляхом перестановок рядків і стовпців (повністю) елемент треба перемістити в правий верхній кут підмасиву. Перевірити чи вийшла на бічній діагоналі спадаюча послідовність елементів.
19. Задано рядок з N^2 цифр. Встановити чи можна, розбивши рядок на підстрічки довжиною N , записати їх у рядки двовимірного масиву $N \times N$ по одній цифрі в одному елементі так, щоб вони в першому стовпці розташувалися в порядку зростання.
20. Знайти мінімальний з неповторюваних елементів двовимірного масиву.
21. Знайти максимальний з повторюваних елементів двовимірного масиву.
22. У двовимірному масиві знайти середнє арифметичне першого стовпця й кількість елементів у кожному з наступних стовпців, що перевищують середнє арифметичне попереднього стовпця.
23. Задано одновимірний масив, який складається з N цілих чисел. Сформувати на його основі двовимірний масив $N \times N$ так, щоб сума елементів у першому стовпці дорівнювала першому елементу одновимірного масиву, сума елементів у другому стовпці повинна дорівнювати другому елементу одновимірного масиву й т.ін. Нулі не використовувати.
24. Визначити скільки елементів двовимірного масиву більші від будь-якого елемента на головній діагоналі.
25. Із двовимірного масиву в одновимірний записали спочатку рядки в довільному порядку, потім стовпці в довільному порядку. Написати програму, яка відновлює вихідний двовимірний масив маючи одновимірний, якщо відомо розмірність двовимірного масиву й елементи в ньому не повторюються.

4. Зміст звіту

1. Постановка завдання.
2. Варіант завдання
3. Текст програми.
4. Результат розв'язку конкретного варіанту.