



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

C PROGRAMMING INTRODUCTION

TUẦN 12: MẢNG VÀ CON TRỎ

Mảng và con trỏ

- Mảng `s` có địa chỉ là phần tử đầu tiên của mảng `s[0]`
- `s` là một con trỏ tới `s[0]`

```
int s[10];  
int *iptr;  
iptr=s; /* From now iptr is equivalent to s */
```
- Cả `iptr` và `s` đều trỏ tới `s[0]`

Con trỏ - mảng

- Mảng là một loại con trỏ!
- Khi mảng được khai báo, một vùng bộ nhớ xác định (có kích thước của mảng) được cấp phát
 - Biến mảng được trỏ tới vị trí đầu tiên của vùng bộ nhớ
- Khi con trỏ được khai báo, nó chưa được khởi tạo
- Giá trị của một biến mảng không thể thay đổi

Thao tác với con trỏ

- Có thể tăng hoặc giảm con trỏ
- Nếu p là một con trỏ có kiểu xác định, $p+1$ trỏ tới địa chỉ của biến tiếp theo có cùng kiểu
- $p++$, $p+i$, và $p += i$ đều có thể được sử dụng

Thao tác với con trỏ

- Nếu **p** và **q** trỏ đến các phần tử trong mảng, **q-p** cho ra số phần tử nằm giữa **p** và **q**.

VD:

```
int main(void)
{
    int a[3] = {17,289,4913}, *p, *q;

    p = a;    /* p points to the beginning of a, that is &a[0] */
    q = p+2;  /* q points to a[2]. Equivalent to q = &a[2]      */

    printf("a is %p\n", a);
    printf("p is %p, q is %p\n", p, q);
    printf("p points to %d and q points to %d\n", *p, *q);
    printf("The pointer distance between p and q is %d\n", q-p);
    printf("The integer distance between p and q is %d\n",
           (int)q-(int)p);
    return 0;
}
```

```
a is 0012FECC
p is 0012FECC, q is 0012FED4
p points to 17 and q points to 4913
The pointer distance between p and q is 2
The integer distance between p and q is 8
```

Truyền mảng cho hàm

- Có thể truyền mảng cho hàm bằng con trỏ
- VD: Hàm tính tổng các phần tử của một mảng

```
#include <stdio.h>
int addNumbers(int *fiveNumber) {
    int i, sum=0;
    for(i=0; i<5; i++, fiveNumbers++) {
        sum+= *fiveNumbers
    }
    return sum;
}
```

Bài tập 12.1

- Viết hàm `countEven(int*, int)` nhận một mảng số nguyên và kích thước của nó và trả về số phần tử lẻ trong mảng

Lời giải

```
int counteven(int* arr, int size) {  
    int i;  
    int count = 0;  
    for (i=0; i<size; i++)  
        if (*(arr+i)%2==0) count++;  
    return count;  
}
```

Bài tập 12.2

- Viết hàm trả về con trỏ trỏ đến số lớn nhất trong một mảng double. Nếu mảng rỗng, trả về NULL.

`double* maximum(double* a, int size);`

Lời giải

```
double* maximum(double* a, int size) {  
    double *max;  
    double *p;  
    int i;  
    max=a;  
    if a==NULL return NULL;  
    for (p=a+1; p<a+size; p++)  
        if (*p > *max) {  
            max = p;  
        }  
    return max;  
}
```

Bài tập 12.3

Viết hàm `getSale` nhận vào tham số là một mảng và số phần tử của mảng là số quý trong năm và yêu cầu người dùng nhập vào doanh số mỗi quý

Viết hàm `totalSale` trả về doanh số trong cả năm

Sử dụng hai hàm này trong chương trình. Sử dụng con trỏ để truyền tham số vào cho hàm

Lời giải

```
#include <stdio.h>

void getSales(float *array, int size){
    int i;
    for(i=0; i<size; i++){
        printf("Enter the sale figure for quarter %d:",
            i+1);
        scanf("%f",array+i);
    }
}

float totalSales(float *array, int size) {
    double sum;
    int i; sum =0;
    for(i=0; i<size; i++){
        sum +=*array;
        array++;
    }
    return sum;
}
```

Lời giải

```
int main()  
{  
    float sales[4];  
    getSales(sales,4);  
    printf("The total sales for the year  
are:    %0.1f\n",totalSales(sales,4));  
    return 0;  
}
```

Bài tập 12.4

- Viết chương trình liệt kê tất cả các mảng con của một mảng. Vd: mảng con của 1 3 4 2 bao gồm:

1

1 3

1 3 4

1 3 4 2

3

3 4

3 4 2

4

4 2

2

Lời giải

```
#include<stdio.h>

void main()
{
    int a[100],n;
    printf("n = "); scanf("%d",&n);
    for(int i=0;i<n;i++)
    {
        printf("\na[%d] = ",i);scanf("%d",&a[i]);
    }
    for(i=0;i<n-1;i++)
    {
        printf("\n%d",a[i]);
        for(int j=i;j<n-1;j++)
        {
            printf("\n");
            for(int k=i;k<=j+1;k++)
                printf("%d\t",a[k]);
        }
    }
}
```


Bài tập 12.5

- Viết chương trình đảo ngược mảng sử dụng:
 - Con trỏ
 - Chỉ số

Lời giải

```
void reversearray(int arr[], int size) {  
    int i, j, tmp;  
    i=0; j= size -1;  
    while(i<j) {  
        tmp=a[i];  
        a[i]=a[j];  
        a[j]= tmp;  
        i++; j--;  
    }  
}
```

Lời giải

```
void reversearray(int *arr, int size) {  
    int i, j, tmp;  
    i=0; j= size -1;  
    while(i<j) {  
        tmp=* (a+i);  
        * (a+i)=* (a+j);  
        * (a+j)= tmp;  
        i++; j--;  
    }  
}
```



25 YEARS ANNIVERSARY
SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**Thank you
for your
attentions!**

