

Mục tiêu:

1. *Hiểu về con trỏ và cấp phát bộ nhớ động.*
2. *Thao tác với kiểu Con trỏ*
3. *Các phép toán trên con trỏ và mảng con trỏ*

Vấn đề 1: Biến con trỏ và khai báo

Chạy đoạn chương trình sau trên máy tính và quan sát kết quả. Giải thích quá trình máy tính cung cấp ô nhớ cho các đối tượng biến nguyên và biến con trỏ nguyên.

```
#include <stdio.h>

void main()
{
    int x=10; int *xtro; xtro=&x;
    printf("\n1. Địa chỉ của biến x là:%p",&x);
    printf("\n2. Địa chỉ của biến xtro là:%p",&xtro);
    printf("\n3. Nội dung của biến x là: %d",x);
    printf("\n4. Nội dung của biến xtro là: %p",xtro);
    printf("\n5. Giá trị mà xtro trỏ đến:%d",*xtro);
    *xtro = 50 ;
    printf("\n6. Giá trị x là %d",x);
}
```

Vấn đề 2: Đối tượng tham chiếu (*) và Phép toán trên con trỏ.

Cả hai toán tử tăng (++) và giảm (--) đều có quyền ưu tiên lớn hơn toán tử tham chiếu (*), vì vậy câu lệnh: *p++; tương đương với *(p++); Cần hiểu:

```
(*p)++ /* đầu tiên lấy giá trị của biến mà con trỏ p trỏ tới, sau đó giá trị của biến mà p trỏ tới tăng lên một đơn vị */
(*p)-- /* đầu tiên lấy giá trị của biến mà con trỏ p trỏ tới, sau đó giá trị của biến mà p trỏ tới giảm xuống một đơn vị */
*(++p) /* đầu tiên con trỏ p trỏ đến địa chỉ kế tiếp, sau đó lấy giá trị chứa trong địa chỉ này */
*(p--) /* đầu tiên lấy giá trị p trỏ tới, sau đó con trỏ p trỏ đến địa chỉ trước nó */
```

Anh/Chị hãy thêm lệnh in giá trị của x sau mỗi lần thay đổi (gán) ở đoạn chương trình dưới đây, ghi lại kết quả khi thực hiện chương trình., Giải thích quá trình máy tính cung cấp ô nhớ và sự thay đổi giá trị của các biến mỗi khi có tác động các phép tính lên các đối tượng đó.

```
#include <stdio.h>

void main()
{
    int a[10],*pa,x;
```

```

a[0]=11; a[1]=22; a[2]=33; a[3]=44;
pa=&a[0];
x=*pa;
pa++;
x=*pa;
x=*pa+1;
x=*(pa+1);
x=**pa;
x=++*pa;
x=*pa++;
}

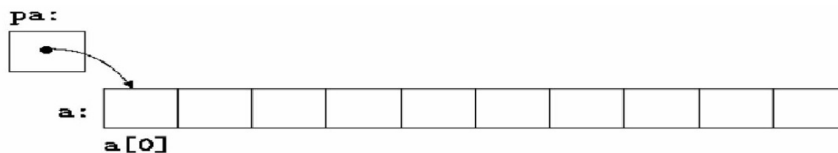
```

Vấn đề 3: Con trỏ mảng một chiều.

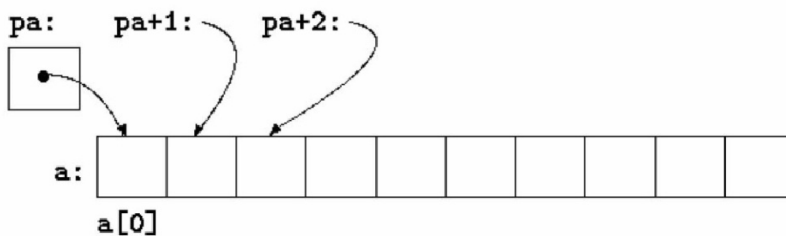
Tên mảng là một con trỏ hằng và trỏ vào phần tử đầu tiên của mảng.

Xét các khai báo sau: `int a[10]; int *pa;`

Với các khai báo này, ta có thể gán `pa = a;` lúc này con trỏ `pa` và `a` đều trỏ đến phần tử `a[0]`.



Và `pa+i` sẽ trỏ tới phần tử `a[i]` như hình vẽ sau:



Lúc này ta có các biểu thức tương đương:

`*pa` tương đương `a[0]`

`*(pa + i)` tương đương `a[i]`

`pa + i` tương đương `&a[i]`

`pa = pa + 4` tương đương `pa = &a[4]`

và `a[i]` có thể thay bằng `*(a + i)`.

Anh/chị hãy chạy các đoạn chương trình sau trên máy tính và giải thích kết quả (trình bày quá trình cung cấp bộ nhớ, và sự thay đổi giá trị ô nhớ sau mỗi câu lệnh).

a) Chương trình 1:

```
#include <stdio.h>
```

```
void main()
{ int a[]={9,8,7,6,18}; int i;
  for(i=0;i<5;i++)printf("%d\n",*(a+i));
}
```

b) Chương trình 2:

```
#include <stdio.h>
void main()
{ int ar[]={10,15,4,25,3,-4}, *p, i;
  p=&ar[2];
  printf("Cac gia tri cua mang:\n\n");
  for(i=0;i<6;i++)printf("ar[%d]=%4d\n",i,ar[i]);
  printf("\nCho biet gia tri cua cac bieu thuc\n\n");
  printf("a) *(p+1)= %d\n\n",*(p+1));
  printf("b) p[-1]= %d\n\n",p[-1]);
  printf("c) (ar-p)=%d\n\n",(ar-p));
  printf("d) ar[*p++]=%d\n\n",ar[*p++]);
  printf("e) *(ar+ar[2])=%d\n\n",*(ar+ar[2]));
}
```

Vấn đề 4: Thao tác trên mảng con trỏ

Giải thích cách thức hoạt động của mảng con trỏ hàm trong chương trình sau đây:

```
#include <stdio.h>
#include <conio.h>
#include <iostream.h>
void nhap(long &x, long &y);
void bp(long &x, long &y);
void lp(long &x, long &y);
void hv(long &x, long &y);
void hienthi(long a, long b);
void main()
{
  const max = 5;
  void (*pfa[max])(long &, long &);
  long a=1, b=2;
  int chon;
  for(int i=0;i<max;i++)
  {printf("\n(1)Nhap (2)Binhphuong (3)Lapphuong (4)Hoanvi \n");
    scanf("%d",&chon);
```

```
switch (chon)
{
    case 1: pfa[i] = nhap;break;
    case 2: pfa[i] = bp;break;
    case 3: pfa[i] = lp;break;
    case 4: pfa[i] = hv;break;
    default:pfa[i] = 0;
}
}
for(i=0;i<max;i++)
{
    pfa[i](a,b);
    hienthi(a,b);
}
getch();
}
void hienthi(long x, long y)
{printf("\n a = %ld, b = %ld \n",x,y);}
void bp(long &x, long &y)
{
    x *= x;
    y *= y;
}
void lp(long &x, long &y)
{
    x = x*x*x;
    y = y*y*y;
}
void hv(long &x, long &y)
{ long tam =x;
  x = y;
  y = tam;
}
void nhap(long &x, long &y)
{ printf("\n Nhap gia tri moi cua a: ");
  scanf("%ld",&x);
  printf("\n Nhap gia tri moi cua b: ");
  scanf("%ld",&y);
}
```