

1 ポインタと一次元配列

学習のポイント

ポインタを用いて一次元配列を参照する方法を学びます。

配列のデータはメモリ上に順序正しく格納されていますので、ポインタを用いて配列の要素を参照することができます。

このように、C ではポインタと配列は密接な関係にあります。

例えば

```
int a[10];
```

という配列があったとき、この配列をポインタで参照するには、ポインタ変数 `pa` を用いて、

```
int *pa;  
pa = a;
```

とします。

配列名 `a` は配列の先頭アドレスを持つポインタ定数と考えることができますので、`pa=a;` により、配列の先頭アドレスが `pa` に代入されます。したがって、`*pa` とすることにより `a[0]` の内容を参照でき、ポインタ `pa` を進めることにより `a[1]`、`a[2]`、・・・の内容を次々に参照することができます。「C 言語」(河西朝雄著 ナツメ社)87 頁

1.1 例題 16

一次元配列のデータをポインタを用いて取り出す。配列データの終わりは-999 とする。

```
/*
  例題 16 C 言語 87 頁
  一次元配列のデータをポインタを用いて取り出す。
  配列データの終わりは-999 とする。
  reidai16.c
*/

#include <stdio.h>

int main()
{
    static int a[] = {10, 35, 25, 67, 88, 29, -999};
    int *pa;

    pa = a;
    while(*pa != -999){
        printf("%d\n", *pa);
        pa++;
    }

    return 0;
}
```

1.2 練習問題 16

例題 16 と同じことを long 型データについて行なうプログラムを作りなさい。なおその際ポインタの値も表示しなさい。

```
/*
  練習問題 16 C 言語 87 頁
  例題 16 と同じことを long 型データについて行なうプログラムを作りなさい。
  なおその際ポインタの値も表示しなさい。
  rensyu6.c
*/

#include <stdio.h>

int main()
{
    static long a[] = {10L, 35L, 25L, 67L, 88L, 29L, -999L};
    long *pa;

    pa = a;
    while(*pa != -999L){
        printf("%04x %ld\n", pa, *pa);
        pa++;
    }

    return 0;
}
```