

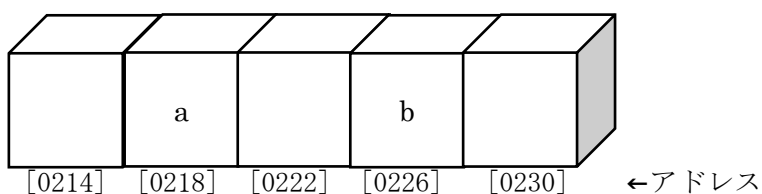
ポインタ

今回解説するのは「ポインタ」及び「アドレス」についてです。
難しいですががんばりましょう。

♣ アドレス

今まで変数 a を宣言したとき、私たちはプログラム上で“a”という文字を用いて変数の呼び出しを行っていました。ですが、実はパソコンの内部では変数“a”の値を保存するためにメモリの一部を使用していました。その変数のメモリ上の位置を「アドレス」と言います。文字通り変数の住所みたいなものです。

例えば二つの int 型の変数 a, b が宣言されているとき、



【メモリのイメージ】

こんな感じでメモリ上の適当な所に変数が割り当てられ、その変数はメモリの位置に対応したアドレスを持ちます。

このアドレスをプログラム上で呼び出したいとき、変数の名前の前に&をつけて表します。

例えば上の図でいうと、&a は 0218 となります。

この辺はあとでまとめるのでなんとなく把握しておいてください。

♣ ポインタ

さて、ここで本題のポインタです。

ポインタとは、先ほどの解説したアドレスを保存できる変数です。

さっきの図でいう 0218 とかを保存する変数ってことです。

宣言は次のように行います。

```
データの型 *変数名(ポインタ名);  
int *p;  
double *q;  
char *s;
```

こんな感じです。

この“*”という文字は、その変数がポインタであるという目印みたいなものです。変数の名前には含まれないので注意してください。

“int*”型の変数“a”を宣言していると考えてください。

➤ ポインタの使用

ポインタを使って変数に代入をすると、次のようになります。

次のサンプルコードを打ち込んでください。

```
#include<stdio.h>

int main(void){
    int a = 810,b = 1919;
    int *p;
    printf("a:%d\n",a);
    p = &a;
    *p = 114;
    printf("a:%d\n",a);
    printf("b:%d\n",b);
    p = &b;
    *p = 514
    printf("b:%d\n",b);
    return 0;
}
```

見慣れない表現が出てきましたね。

*p というのは「ポインタ p が指す変数」を指します。

p は「変数のアドレス」を表すのに対し、*p は「p が指すアドレスの変数の中身」です。

このサンプルコードでは7行目にポインタ p には a のアドレスを入れているので、その間は*p と書くのと a と書くのは同じ意味になります。

11行目ではポインタ p に b のアドレスが代入されています。

このとき*p と書くのと b と書くのは同じ意味になります。

「ポインタ p の中に入っているアドレスの値によってポインタ p が示す変数が変わる」ことを意識してください。このソースコードではポインタ p の中身を変えてポインタ p だけで2つの変数に代入を行っています。

➤ 宣言まとめ

もうお分りの通り、ポインタは宣言含め色々複雑です。

次に例として int 型を指すポインタを用いたいろんな操作をまとめます。

【宣言】

```
int *p;
```

【代入】

```
int a;
```

```
p = &a; // 変数に&をつけるとその変数のアドレスを表します。
```

```
*p = 30; // この時、*p と書くことと a と書くことは一緒です
```

```
a = 30; // つまり、この行と 1 行前は全く同じ代入を行っています。
```

➤ ポインタを引数とした関数

前は関数を学びましたが、今日学んだポインタと組み合わせて関数の外から値を変えたり変えなかつたりできます。

とりあえず次のサンプルコードを打ち込んでみましょう。

```
#include<stdio.h>

void nibai(int *p){
    *p *= 2;
}

int main(void){
    int a = 100;
    printf("%d¥n", a);
    nibai(&a);
    printf("%d¥n", a);
}
```

関数 nibai に a のアドレスを送ってあげると main 文で a の値を変化させていないのに変化していることがわかりますね。これは関数 nibai でメモリに保存されている値を直接変化させたからです。

このように、ポインタを使って関数の外から関数の値を変化させることができます。

それでは！練習問題！いってみましょう！！！！！！！！！！！！！！！！！！

❖ 練習問題

1. 大きさ 10 の配列を宣言して、その要素のアドレスを全て各行に表示させてください。
表示には%p を使ってください。(今まで%d と書いていた場所を%p に変えれば ok)
2. 変数のアドレスを受け取ってその変数の値を絶対値の値に変える関数
void absolute(int *p)を作ってください。