

入門講習会 文字

1. 文字コード

文字はコンピュータ内部では数値として扱われている。つまり、ある文字とある整数値には対応関係がある。文字'a'にはそれに対応する整数があり、文字'7'にもそれに対応する整数がある。

文字と数値の対応関係を「文字コード」と呼ぶ。文字コードにも様々な種類がある。どの文字コードが使われているのかは環境依存である。しかし英数字の場合、多くは「ASCII コード」と呼ばれる文字コードに則している。以下、英数字は ASCII コードに則しているものとして話を進める。

実際にプログラムを組んで、文字と数値の対応関係を見てみよう。以下は、'A' 'B' 'C' という文字を取って%d で表示してみるプログラムである。

```
#include <stdio.h>

int main(void)
{
    printf("%d %d %d", 'A', 'B', 'C');
    return 0;
}
```

実行結果は次のようになる。

```
65 66 67
```

この結果から、'A'は 65, 'B'は 66, 'C'は 67 という数で扱われていることが分かる。

A,B,C は数値としてみると連続している。実際、ASCII コードにおいては、アルファベット A~Z は数値 65~90 と対応している。そこで、大文字アルファベットをすべて表示するプログラムは次のように書ける。

```
#include <stdio.h>

int main(void)
{
    int i;
    for(i = 0; i < 26; i++) {
        printf("%c", 'A' + i);
    }
    return 0;
}
```

逆に、大文字アルファベットが A から数えて何番目(0 番目から数える)かを出力するプログラムは次のように書ける。

```
#include <stdio.h>

int main(void)
{
    char c;
    scanf("%c", &c);
    printf("%d\n", c - 'A');
    return 0;
}
```

実行結果は以下のようになる。

```
R (入力)
17
```

大文字アルファベットの例を示したが、小文字アルファベットの場合も勿論同様に書ける。上のソースコードにおいて、'A'の部分をもっと'a'とすればよい。

文字は数値として扱われるのだが、ある文字 c が大文字アルファベットであるかを判定するには、次のように書けば良い。

文字としての数字を、整数値としての数字に変換することができる。このことを利用すると、ABC081_A の解答が次のようなコードで書ける。

```
#include <stdio.h>

int main(void)
{
    char S[5];
    int ans;

    scanf("%s", S);
    ans = (S[0] - '0') + (S[1] - '0') + (S[2] - '0');
    printf("%d\n", ans);

    return 0;
}
```

2. char 型

文字は内部的には整数値であることを、前項で述べた。char 型の変数は文字を入れるための変数として利用してきたが、文字が整数と同じなら、char 型の変数は int 型のように、整数値を扱えると考えられる。実際その通りで、char 型も int 型もその内部表現は同じで、異なるのはそのサイズだけである。ただし、それが符号を扱えるのかどうかは処理系定義のため注意が必要である。符号についての詳しい話は別資料で述べる。

3. 文字列走査の一例

文字コードとは少し話がそれるが、文字列走査の例を示そう。

以下は、入力された文字列が回文であるかを調べるプログラムである。

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

int main(void)
{
    char str[100];
    int flag = 1;
    int l, r;
    scanf("%s", str);

    l = 0;
    r = strlen(str) - 1;
    while (l < r) {
        if (str[l] != str[r]) {
            flag = 0;
            break;
        }
        l++;
        r--;
    }
    if(flag) printf("Yes\n");
    else printf("No\n");

    return 0;
}
```

注目したいのは、左から走査する変数 r と、右から走査する変数 l があることである。2つのカウンタを両端から内側へ動かして調べている。このように、配列を操作するた

めに、カウンタを二つ用意してやるとうまく処理が書けることがあるので、覚えておくとよいだろう。