

目次

第1章 第4回放送	3
1.1 第4回放送内容	3
1.2 if構文	3
1.2.1 条件演算子	3
1.3 switch構文	4
1.3.1 switch構文とは	4
1.3.2 breakを書かないswitch構文	5
1.4 do while for構文	6
1.4.1 do while構文	6
1.4.2 while構文	7
1.5 for構文	8
1.5.1 for構文の書き方	8
1.5.2 forを使った無限ループ	10
1.6 演習問題	10
1.6.1 文字列をコピーするプログラム	10
1.6.2 文字数カウントプログラム	12
1.6.3 最大値を求めるプログラム	12
1.6.4 九九表を作成するプログラム	13
1.7 本日の講義のおさらい	14

第1章 第4回放送

1.1 第4回放送内容

1. if 構文
2. switch 構文
3. do while for 構文
4. 演習問題

1.2 if 構文

前回の講義で if 構文を勉強しました。if とは、もし～ならば A の処理を、そうでなかったら B の処理をするといった場合に使うんですね。ここで、if の別の書き方を紹介します。

1.2.1 条件演算子

あまり使わないほうがよいですが、if 文には別の書き方が存在しまして条件演算子というものが存在します。ここはさらっと流してもらって結構です。

書式

条件 ? 真のときの処理 : 偽のときの処理

このように書きます。これだけでは何のことだかわからないと思いますから、例を挙げてみます。前回の放送の演習問題で絶対値を求めるプログラムを作成せよ、という問題がありました。良い例ではありませんが、これを条件演算子を使って書き直してみますと、次のようになります。

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

```
int main(void)
{
    int x;

    printf("数字を入力してください:");
    scanf("%d",&x);

    x = ( x < 0 ) ? -x : x;

    printf("絶対値は%d です\n",x);
    return 0;
}
```

このように使います。

1.3 switch 構文

1.3.1 switch 構文とは

さて、if 構文をマスターしたところで、次は switch 構文を学んでみましょう。switch 構文とは if 構文を少し拡張したものだと考えてもらえれば結構です。ここで、前回の放送の if 構文の例を書きます。

```
int hoge = 8;

if( hoge == 10 ){
    //A の処理
}else if( hoge == 9) {
    //B の処理
}else if( hoge == 8 ) {
    //C の処理
}else{
    //D の処理
}
```

こんな感じの例がありましたね。しかし、ここで hoge が 7 である場合、hoge が 6 である場合…と考えると、なんだか面倒な文になってしまいますね。これを解消するのが switch 構文です。ここで switch 構文を使って書き直してみましょう。

```
int hoge = 8;

switch( hoge ) {
case 10: //処理 A
break;
case 9 : //処理 B
break;
case 8 : //処理 C
break;
default: //処理 D
break;
}
```

switch に続くカッコの中には何か式を書きます。この場合 hoge と書いただけなので、hoge の中の値によって処理を分けます。ここで、例えば hoge の値が 8 であった場合、case 8: という場所に飛びます。default というのはそのほかの場合であった場合、if でいう else のような場合を表しています。そして break というものが存在しますが、break とは switch 構文から飛び出るということを意味しています。

1.3.2 break を書かない switch 構文

switch 構文の中で break を書かないとどうなるでしょうか。次のコードを実行してみましょう。

```
#include <stdio.h>

int main(void)
{
```

```
int hoge = 9;

switch( hoge ) {
case 10: printf("10\n");

case 9 : printf("9\n");

case 8 : printf("8\n");

default: printf("7\n");

}
return 0;
}
```

コンパイルして実行してもらえばわかると思いますが、C言語は上から下へコードが実行されるので、そのまま下の文が実行されます。このようなテクニックも使われることがあります。

1.4 do while for 構文

例えば”Hello”と10回表示したい場合があるかもしれません。そういった場合に使われるのがdo while for 構文です。つまり、何か処理を繰り返し行いたい場合使うものです。

1.4.1 do while 構文

ここで、hello vip と10回表示するプログラムをdo while を使って作ってみます。

```
#include <stdio.h>

int main(void)
{
int hoge = 10;
```

```
do{
printf("%d回目:Hello vip\n", 11 - hoge);
hoge--;
}while( hoge > 0);

return 0;
}
```

ここで、do while 構文の書式を書きます。

書式

```
do { 処理 } while( 条件式 );
```

よく忘れるのですが、while の後にセミコロンを付けることに注意してください。do の中に何か処理を書きます。そして条件式には条件を書き、この条件が真の場合はまた do のはじめの部分に戻って処理を繰り返します。

1.4.2 while 構文

do while より while だけの構文のほうがよく使われます。do while と while の違いは、条件式が最初に判定させるかあとで判定させるかの違いだけです。Hello vip と 10 回表示するプログラムを while を使って書き直すと次のようになります。

```
#include <stdio.h>

int main(void)
{
int hoge = 10;

while( hoge > 0 ) {
printf("%d回目:Hello vip\n",11-hoge);
hoge--;
}
```

```
return 0;
}
```

書式は次のようになります。

```
書式
while( 条件式 ) { 処理 }
```

ここで、条件式はifと同じです。ifと同じということは次のようにしたらどうなるのでしょうか。

```
while( 1 ) {
printf("Hello vip\n");
}
```

これは、常に真ということを表しています。つまり無限ループに陥ります。

1.5 for 構文

for 構文は while 構文とあまり変わりません。

1.5.1 for 構文の書き方

while で使った例を for を使って書き直してみます。

```
#include <stdio.h>

int main(void)
{
int hoge;
```



```
for( hoge = 10; hoge > 0 ; hoge-- ) {  
printf("%d 回目:Hello vip\n",11-hoge);  
}  
  
return 0;  
}
```

このようになります。

書式

```
for( 初期条件 ; 条件式 ; 後始末 ) { 処理 }
```

このようになります。初期条件や後始末、条件式などは省くことができます。例えば、

```
int hoge = 10;  
  
for( ; hoge > 0 ; hoge-- ) {  
printf("%d 回目:Hello vip\n",11-hoge);  
}
```

とすることもできます。for と while を比較してみましょう。

while の場合

```
処理 A ;  
  
while( 処理 B ) {  
  なにかしたい処理  
  
    処理 C ;  
}
```

これを for で書き直すと、

for の場合

```
for( 処理 A ; 処理 B ; 処理 C ) {  
    なにかしたい処理  
}
```

1.5.2 for を使った無限ループ

for を使った無限ループは簡単に行うことができます。それは条件式になにも書かなければよいのです。

```
for(;;){ なにか処理 }
```

1.6 演習問題

1.6.1 文字列をコピーするプログラム

仕様

キーボードから英単語を受け取り、その英単語を別の配列にコピーする

```
//文字列コピープログラム 1
```

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

```
int main(void)
```

```
{
```

```
char source[100] , str[100];
```

```
int i;
```

```
printf("英単語を入力してください:");
```

```
scanf("%s",&source[0]);
```

```
for( i = 0; source[i] != '\0' ; i++ ){
    str[i] = source[i];
}
str[i] = '\0';

printf("コピーした文字列は %s です",str);
return 0;
}
```

別の方法も存在します

//文字列コピープログラム 2

```
#include <stdio.h>

int main(void)
{
    char source[100] , str[100];
    int i;

    printf("英単語を入力してください:");
    scanf("%s",&source[0]);

    i = 0;
    while( source[i] != '\0' ) {
        str[i] = source[i];
        i++;
    }
    str[i] = '\0';

    printf("コピーした文字列は %s です",str);
    return 0;
}
```

1.6.2 文字数カウントプログラム

仕様

キーボードから英単語を受け取り、その文字数を数える
ヒント：文字列の終わりには必ずヌル文字があります。

//文字数カウントプログラム

```
#include <stdio.h>

int main(void)
{
    char str[100];
    int len;

    printf("何か英単語を入力してください:");
    scanf("%s",&str[0] );

    for( len = 0; str[len] != '\0' ; len++ ){

    printf("文字数は%d です\n",len);
    return 0;
    }
```

1.6.3 最大値を求めるプログラム

仕様

キーボードから 10 個の数字を受け取り、その最大値を表示する

//最大値を求めるプログラム

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

```
int main(void)
{
int array[10];
int i,j,max;

printf("10個の数字を入力してください\n");

for( i=0 ; i < 10 ; i++ ) {
printf("%d回目:",i+1);
scanf("%d",&array[i]);
}

for( j=1 , max = array[0] ; j < 10; j++) {
if( max < array[j] ) {
max = array[j];
}
}

printf("一番大きな数字は%dです",max);
return 0;
}
```

1.6.4 九九表を作成するプログラム

仕様

九九を表示するプログラムを作成せよ。表示結果は以下のようになる。

```
1  2  3  4  5  6  7  8  9
2  4  6  8 10 12 14 16 18
3  6  9 12 15 18 21 24 27
4  8 12 16 20 24 28 32 36
5 10 15 20 25 30 35 40 45
6 12 18 24 30 36 42 48 54
7 14 21 28 35 42 49 56 63
8 16 24 32 40 48 56 64 72
9 18 27 36 45 54 63 72 81
```

```
#include <stdio.h>

int main(void)
{
    int i,j;

    for( i=1 ; i < 10 ; i++ ) {
        for(j=1 ; j != 10 ; j++ ) {
            printf("%3d ",i*j);
        }
        printf("\n");
    }
    return 0;
}
```

ここで、printfの中に%3dというものを使ったが、これは3文字分のスペースを確保して表示するという意味である。このようにすると綺麗に表示することができる。5文字分のスペースを確保したい場合は、%5dとすればよい。

1.7 本日の講義のおさらい

- if 構文
条件演算子というものが存在する
- switch 構文
プログラムの流れを分岐するときに使うと便利である。
- do while for 構文
繰り返しの処理をする場合に使う

参考文献

- [1] Brian W. Kernighan, Dennis M. Ritchie: The C Programming Language Second Edition.
- [2] ハーバートシルト著 トップスタジオ訳: 独習C 第三版.
- [3] 田中 敏幸著 C言語によるプログラミングの基礎
- [4] 柴田 望洋 明解C言語
- [5] 松本 健一 上田 悦子 安室 喜弘 2006年度プログラミング演習(初級コース)課題

<http://chihara.naist.jp/people/STAFF/yasumuro/Pub/c-ensyu2006/>
- [6] Ryo Kawahara C/C++プログラミング初心者講座

http://www.stat.phys.kyushu-u.ac.jp/~ryokawa/cbegin2_3/pdf/cbegin.pdf
- [7] TOMOJI 初心者のためのポイント学習C言語

<http://www9.plala.or.jp/sgwr-t/>
- [8] 小出 俊夫 C言語入門インターネット版

<http://homepage1.nifty.com/toshio-k/prog/c/>
- [9] 赤坂 玲音 C言語入門

<http://wisdom.sakura.ne.jp/programming/c/index.html>