

C言語講座

第2回

第2回の内容 ♪ ♪

- 関数
- 再帰プログラム
- ファイルの入出力

関数とは？

- 関数・・・今まで使ってきたmain関数やprintf,scanf・・・etcのことです。
 - 今までは、main関数の中にプログラムを書いてきました。
- 今回は「自分で作る関数」についてやります。

自作関数について

- 自作関数の作り方

```
返却値型 変数名(引き数){  
    関数の内容;  
    return 戻り値;  
}
```

返却値型・・・main関数に戻したときの値の型を指定している。

引き数・・・main関数から呼び出すときに持ってくる値の型を宣言しています。

※返却値型がvoid型のときは、引き数はいらぬ。

自作関数の使用例

```
#include<stdio.h>
int seki(int a,int b){
    int c;
    c=a*b;
    return c;
}

void main(){
    int x,y,z;
    printf("x=");
    scanf("%d",&x);
    printf("y=");
    scanf("%d",&y);
    z=seki(x,y);
    printf("%d*d=%d¥n",x,y,z);
}
```

自作関数

※自作関数では二つの値を掛け算しています。

main
関数

自作関数説明

```
#include<stdio.h>

int seki(int a,int b){
    int c;
    c=a*b;
    return c;
}

void main(){
    int x,y,z;
    printf("x=");
    scanf("%d",&x);
    printf("y=");
    scanf("%d",&y);
    z=seki(x,y);
    printf("%d*%d=%d¥n",x,y,z);
}
```

- 自作関数はmain関数の前に書く

自作関数のseki関数はmain関数のここで呼ばれています。

- main関数で入力したx,yの値がseki関数のa,bにそれぞれに入られ、zにseki関数からの返却値cが代入されます。

例題

入力した二つの値を、自作関数で四則演算と余剰(%のこと)をもとめるプログラム作成してください。

ヒント

自作関数は5つ必要です。

例題の答え1

```
#include<stdio.h>
int seki(int a,int b){
    int c;
    c=a*b;
    return c;
}
int wa(int a,int b){
    int c;
    c=a+b;
    return c;
}
int hiki(int a,int b){
    int c;
    c=a-b;
    return c;
}
```

```
int zyo(int a,int b){
    int c;
    c=a/b;
    return c;
}
int yo(int a,int b){
    int c;
    c=a%b;
    return c;
}
```

例題の答え2

```
void main(){
    int x,y,z;
    printf("x=");
    scanf("%d",&x);
    printf("y=");
    scanf("%d",&y);
    z=seki(x,y);
    printf("%d*%d=%d\n",x,y,z);
    z=wa(x,y);
    printf("%d+%d=%d\n",x,y,z);
    z=zyo(x,y);
    printf("%d/%d=%d\n",x,y,z);
    z=hiki(x,y);
    printf("%d-%d=%d\n",x,y,z);
    z=yo(x,y);
    printf("%d%%%d=%d\n",x,y,z);
}
```

プロトタイプ宣言

- プロトタイプ宣言とは？

今まで、自作関数は、main関数の前に書いていたが、main関数の前に関数名を宣言して、main関数の後ろに内容を書くことができます。

例文

```
#include<stdio.h>

int seki(int a,int b);

void main(){
    int x,y,z;
    printf("x=");
    scanf("%d",&x);
    printf("y=");
    scanf("%d",&y);
    z=seki(x,y);
    printf("%d*%d=%d",x,y,z);
}

int seki(int a,int b){
    int c;
    c=a*b;
    return c;
}
```

プロトタイプ宣言

- プロトタイプ宣言の後ろにはセミコロンをつけてください
- この様にあとから関数の内容を書きます。

再帰関数

再帰関数とは？

関数内で自分自身の関数を呼び出すことを言います。

注)再帰関数を終了させる条件を設定しておかないと、永遠に自分を呼び続けることになるので、気を付けましょう。

例文

```
#include<stdio.h>
void raretu(int);

void main(){
    int a;
    printf("整数を入力してください");
    scanf("%d",&a);
    raretu(a);
}

void raretu(int x){
    if(x<100){
        printf("%d¥n",x);
        raretu(x+1);
    }
}
```

再帰条件を抜けるためには、この再帰関数に持ってきた引き数が100未満のとき、引き数に1を加えて、ふたたび自分自身を呼び込む、引き数が100以上であるとき、関数を抜ける

演習問題

整数を入力し、入力された n の階乗を求めるプログラムを生成せよ。

ヒント

(階乗の定義)

$n=0$ ならば $0!=1$

$n \neq 0$ ならば $n!=n*(n-1)!$

If文で分岐

答え1

```
#include<stdio.h>
int factorial(int n);

void main(){
    int n,result;
    printf("整数を入力してください");
        scanf("%d",&n);
        result=factorial(n);
        printf("%dの階乗は%dです¥n",n,result);
}
int factorial(int n){
    int i,result=1;
    for(i=n;i>0;i--){
        result*=i;
    }
    return result;
}
```

答え2 (再帰関数使用例)

```
#include<stdio.h>
int factorial(int n);

void main(){
    int n,result;
    printf("整数を入力してください");
        scanf("%d",&n);
        result=factorial(n);
        printf("%dの階乗は%dです¥n",n,result);
}
int factorial(int n){
    if(n==0){
        return 1;
    }
    else{
        return n*factorial(n-1);
    }
}
```

ファイルの入出力

ファイルとモード

- ファイルには「テキストファイル」と「バイナリファイル」の2種類があり、どちらのファイルをどのような形で開くのかを指定するのが、モードの役割です。
- テキストファイルはごく一般的なファイルであるが、バイナリファイルでは音声や画像を扱うことが可能です。

※テキストファイルについてのみ解説します。

システム標準関数

- システム標準関数とは？

“printf”や“scanf”はMicrosoft Visual C++ 2010によって備え付けられた関数のことである。

- このシステム標準関数を利用することによって、キーボードからの入力や画面への出力に限らず、ファイルの入出力も可能になる。

ファイル入出力のシステム関数

- fopen : ファイルを開く。
- fclose : ファイルを閉じる。
- fprintf : ファイルに書式付の文字列を書き込む(書き込みモードに使用される)
- fscanf : ファイルから文字列を読み込み、指定された形式に変換する。(読み込みモードに使用される)
- fgets : ファイルから1行読み出す。
- fputs : ファイルに1行読み出す

他にもいくつか存在する。モードについては後程

ファイルオープン・クローズの書き方

- ファイルオープン

```
FILE *変数名;
```

```
変数名=fopen(“ファイル名”, “モード”);
```

- ファイルクローズ

```
fclose(変数名);
```

説明

FILE ***変数名**;

FILE*型ポインタ変数の宣言。

変数名=fopen(“**ファイル名**”, “**モード**”);

ファイルを開く操作の記述。

fclose(**変数名**);

開いたファイルを閉じる操作の記述。

モードの種類

- r: テキストファイルの読み込みモード。
- w: テキストファイルの書き込みモード。
- a: テキストファイルの追加モード。
- r+: テキストファイルの更新モード。

例文 I

```
#include <stdio.h>
void main(){
    FILE *fp;
    fp=fopen("open.txt","w");
    if(fp==NULL){
        printf("ファイルオープン失敗。¥n");
    }
    fclose(fp);
}
```

- ファイル「open.txt」がフォルダの中に作られているはず。
- 4行目の"w"を"r"や"a"にすると？

例文Ⅱ

```
#include <stdio.h>
void main(){
    FILE *fp;
    char str[1024];
    fp=fopen("ta.cpp","r");
    if(fp==NULL){
        printf("ファイルが存在しません。");
    }
    while((fgets(str,1024,fp))!=NULL){
        printf("%s",str);
    }
    fclose(fp);
}
```

例文 I、II の補足

- 例文 I

4行目のかっこ内を「"close.txt","r"」にしてデバッグすると、エラーが出力される。が「"close.txt","a"」だと、フォルダに新しくclose.txtが作られる。先に「"close.txt","a"」をデバッグし、「"close.txt","r"」でデバッグすると、何も起こらず終了する。

- 例文 II

存在するファイル名であれば、そのファイルが読み込まれる。存在しないファイル名ならば、エラーで返される。

演習 I

- 例文 I で作った「open.txt」に、次のスライドのテンプレを書いてください。
書いたら保存させてください。
- その自己紹介をデバッグ画面と「close.txt」に同時に表示させるプログラムを生成してください。

テンプレ

学部:

学科:

名前:

生年月日:

出身地:

.txt以外には

- 先ほどは.txtファイル(メモ帳)を扱う入出力を紹介しましたが、ワードやエクセルにも入出力させることができる。
- ここからは例として、エクセルに出力させるプログラムを学びたいとおもいます。

エクセルにファイルを作るには

- 「ファイル名 **.CSV**」←このようにしてください。
- 試しに、メモ帳を起動し、適当に文章や数字を入力した状態で名前を付けて、.txtを.csvに変えて保存してみよう。

改行とセルの移動

- .改行はエクセルでもそのまま改行されて出力される。
- .ただし、このままでは1個のセルにすべて入力されてしまう。隣のセルにも出力させるには、データ間を「,」で区切る必要がある。

例文

```
#include<stdio.h>
int main(void){
int x=12,y=15,z=19;
FILE*fp;
fp=fopen("4-4.csv","w");
if(fp==NULL)
printf("ファイルはありません\n");
else{
fprintf(fp,"x,%d\n",x);
fprintf(fp,"y,%d\n",y);
fprintf(fp,"z,%d\n",z);
fclose(fp);}
return 0;
}
```

例文補足

- ・先ほどと同様に、プログラムによって出力させることができる。
- ・「,」によって区切ることで隣のセルへの書き込みが可能になり、「¥n」によって改行させることができる。

演習問題

- .5科目の点数を入力し、それぞれの点数、合計、平均をエクセルに表として出力させ、平均点数によって何らかの評価をエクセルに表示させるプログラムを作成してください。

ヒントコード1

```
#include <stdio.h>
int main(void)
{
FILE *fp;
char *fname = "ファイル名と拡張子は?";
char kamoku[5][7]={"国語","数学","英語","理科","社会"};
char hyouka[5][16]={"神","優等生","凡人","マダオ","脳筋"};
int sum=0;
double ave=0;
int i;
int a[5];
for(i=0;i<5;i++){
printf("%sの点数を入力してください:",&kamoku[i][0]);
scanf("%d",&a[i]);
sum+=a[i];
}
ave=sum/5.0;
```

ヒントコード²

```
fp = fopen( fname, "モードは何か？" );
if( fp == NULL ){
printf( "%sファイルが開けません\n", fname );
return -1;
}
for(i=0;i<5;i++){
fprintf( fp, "%s,%d\n",&kamoku[i][0],a[i]);
}
fprintf( fp, "sum is,%d\n",sum);
fprintf( fp, "average is,%lf\n",ave);
if(ave>=90){fprintf(fp,"評価...,%s",&hyouka[0][0]);}
if(ave>=80&&ave<90){fprintf(fp,"評価...,%s",&hyouka[1][0]);}
if(ave>=70&&ave<80){fprintf(fp,"評価...,%s",&hyouka[2][0]);}
if(ave>=60&&ave<70){fprintf(fp,"評価...,%s",&hyouka[3][0]);}
if(ave<60){fprintf(fp,"評価...,%s",&hyouka[4][0]);}
ここにファイルを閉じる処理を記述しよう
printf( "%sファイル書き込みが終わりました\n", fname );
return 0;
}
```