



2007年10月12日

## 第6回 C言語によるプログラミング (3)

# 情報処理論 (応用)

松山大学 経営学部

檀 裕也

<http://www.cc.matsuyama-u.ac.jp/~dan/education/application/>



# 前回の実習課題

- 成績評価のプログラムを作成し、ソースコード `score.c` を添付してメールを送信せよ。
  - 宛先: `dan@cc.matsuyama-u.ac.jp`
  - 件名: プログラミング課題#04

```
C:¥>score.exe  
点数 : 90  
成績は A です。
```

```
C:¥>
```

```
C:¥>score.exe  
点数 : 65  
成績は C です。
```

```
C:¥>
```



# 解答例

```
#include <stdio.h>

int main()
{
    int x;

    printf( "点数：" );
    scanf( "%d", &x );

    if( x >= 80 ){
        printf( "成績は A です。" );
    }
    else if( x >= 70 ){
        printf( "成績は B です。" );
    }
    else if( x >= 60 ){
        printf( "成績は C です。" );
    }
}
```

```
else{
    printf( "成績は X です。" );
}

return 0;
}
```



# 解答例 (その2)

```
#include <stdio.h>


int main()
{
    int x;

    printf( "点数：" );
    scanf( "%d", &x );

    if( x < 0 ){
        printf( "入力エラー\n" );
    }
    else if( x > 100 ){
        printf( "入力エラー\n" );
    }
    else if( x >= 80 ){
        printf( "成績は A です。 \n" );
    }
}
```

```
    else if( x >= 70 ){
        printf( "成績は B です。 \n" );
    }
    else if( x >= 60 ){
        printf( "成績は C です。 \n" );
    }
    else{
        printf( "成績は X です。 \n" );
    }

    return 0;
}
```



# 解答例 (その3)

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

```
int main()
```

```
{
```

```
    int x;
```

```
    printf( "点数：" );
```

```
    scanf( "%d", &x );
```

```
    switch( x / 10 ){
```

```
        case 10:
```

```
        case 9:
```

```
        case 8:
```

```
            printf( "成績は A です。¥n" );
```

```
            break;
```

```
        case 7:
```

```
            printf( "成績は B です。¥n" );
```

```
            break;
```

```
        case 6:
```

```
            printf( "成績は C です。¥n" );
```

```
            break;
```

```
        default:
```

```
            printf( "成績は X です。¥n" );
```

```
            break;
```

```
    }
```

```
    return 0;
```

```
}
```



# 今回の授業内容

---

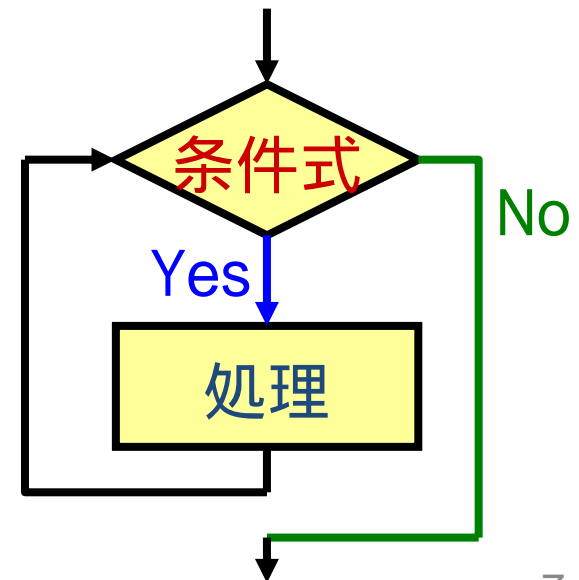
- C言語によるプログラミング (3)
  - while 文による繰り返しの処理



# while 文

- **条件式**が成立している間、ブロック内の**処理**を繰り返す。
  - 条件式の判定      処理の実行
  - 初期条件によっては処理を実行しないことがある

```
while( 条件式 ){  
    処理  
}
```





# 例題 1

study.c

```
#include <stdio.h>

int main()
{
    int x, score = 30;

    while( score < 60 ){
        printf( "現在のスコアは%dです。¥n", score );
        printf( "勉強量：" );
        scanf( "%d", &x );
        score = score + x;
    }

    printf( "合格点に到達しました。" );
    printf( " (スコア%d)¥n", score );

    return 0;
}
```





# 実行例

- 合格点（60点）に達するまでひたすら勉強する

```
C:¥>study.exe
```

現在のスコアは30です。

勉強量：10

現在のスコアは40です。

勉強量：15

現在のスコアは55です。

勉強量：10

合格点に到達しました。（スコア65）



# 解説

- while 文の条件式は **score < 60**
- 処理の中で変数 score の値が変化し、条件式を満足しなくなったらループを終了する。

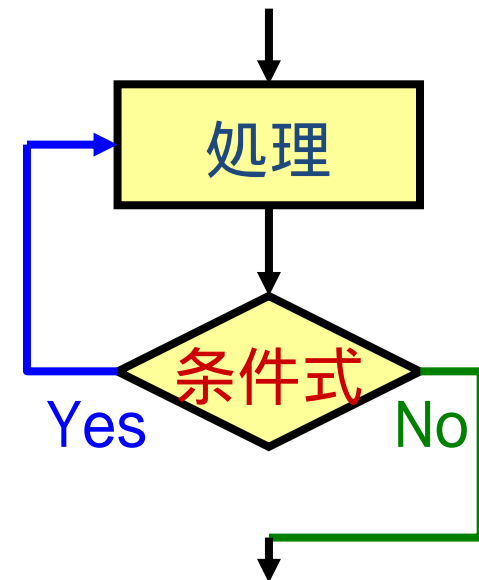
```
while( score < 60 ){  
    printf( "現在のスコアは%dです。¥n", score );  
    printf( "勉強量：" );  
    scanf( "%d", &x );  
    score = score + x;  
}
```

score に x の値を加える



# do ~ while 文

- ブロック内の**処理**を実行し、**条件式**が成立していれば処理を繰り返す。
  - 処理の実行      条件式の判定
  - 少なくとも1回は処理を実行する





## 例題 2

- 2桁の整数を入力させる (digit.c)

```
#include <stdio.h>
int main()
{
    int x;

    do{
        printf( " 2桁の整数 : " );
        scanf( "%d", &x );
    }while( x < 10 || 99 < x );

    printf( "x = %d\n", x );

    return 0;
}
```

2桁の整数が入力されるまで  
繰り返し整数を受け取る。

```
C:¥>digit.exe
2桁の整数 : 1
2桁の整数 : 100
2桁の整数 : 10
x = 10

C:¥>
```



# 論理演算子

- if 文や while 文の条件式において、複数の条件式を組み合わせたことができる。
  - 論理積 (AND) :      `&&`      `0 <= x && x < 20`
  - 論理和 (OR) :      `||`      `x < 10 || 99 < x`
  - 否定 (NOT) :      `!`      `!( x > 0 )`



# 解説

- do 文のブロック内の処理を実行し、条件式の真偽を判定する。
  - 入力された整数が2桁のときはループを終了する
  - そうでないときは、再入力を求める

```
do{  
    printf( " 2桁の整数：" );  
    scanf( "%d", &x );  
}while( x < 10 || 99 < x );
```



# 実習課題

- 合計計算のプログラムを作成し、ソースコード `sum.c` を添付してメールを送信せよ。
  - 宛先: `dan@cc.matsuyama-u.ac.jp`
  - 件名: プログラミング課題#05

データ0が入力されたら合計を表示してプログラムを終了する

```
C:¥>sum.exe  
データ : 10  
データ : 12  
データ : 15  
データ : 0  
合計は 37 です。
```



# 次回の予定

---

- 日時： 10月17日（水）  
5時限
- C言語によるプログラミング（4）  
– 繰り返しの処理