

システム科学科 情報処理演習テキスト 4  
- 制御構造 3: 条件分岐と繰り返しの組み合わせ -

原案:佐藤宏介 修正: 升谷 保博, 庄野 逸, 鳩野逸生, 橋本守, 才脇直樹, 田中宏喜, 日浦慎作

ポイント 1: 繰り返し処理の中で条件分岐を使うことができる。

example4\_1.c

```
/*
 * 白黒円の市松模様を描く
 */
#include <stdio.h>
#define IXMAX 10
#define IYMAX 7
int main(void)
{
    int ix, iy;

    /* 格子状に と を並べる */
    for ( ix = 0; ix < IXMAX; ix++ ) {
        for ( iy = 0; iy < IYMAX; iy++ ) {
            /* 市松模様になるように と を切替え */
            if ( (ix+iy)%2 == 0 ) {
                printf(" ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

ポイント 2: 繰り返しを途中で抜けるには break 文を用いる。繰り返し構造が入れ子になっている場合の break 文では、一番内側の繰り返しを抜ける。

【例】

```
for( i = 0 ; i < 100 ; i++ ){
    a = 3.1 * i;
    if( a > 100.0 ){
        /* for ループから抜ける */
        /* 次は Exit Loop と表示する行へ */
        break;
    }
    b = 2.4 * a;
}
printf( "Exit Loop\n" );

while( i < 100 ){ /* 繰り返し 1 */
    ...
    for( j = 0; j < 200; j++ ){ /*繰り返し 2 */
        ...
        if( a > 3.0 ){
            break; /* 繰り返し 2 を抜ける */
        }
        ...
    }
    printf( "Exit 繰り返し 2\n" );
}
printf( "Exit 繰り返し 1\n" );
```

ポイント 3: 条件分岐 (if, for, while, do while) の条件は, "0" (偽を表す) と "0 以外" (真を表す) の値で制御されている

【例】

```
i = 0;
while(1){ /* 1 は真を表すので, 常に真. 無限ループとなる*/
    if( i > 10){
        break; /* 無限ループを抜ける
    }
    i++;
}
```

example4\_2.c

```
/*
 * 無限ループと, break, 関係演算子
 */
#include <stdio.h>
int main(void)
{
    int n;
    while(1){ /* 偽は 0, 真は 0 以外として制御されるため, 無限ループとなる*/
        printf("何か数値を入力して下さい (0 以外を入力すると, 終了します): ");
        scanf("%d",&n);
        if(n){ /* 偽は 0, 真は 0 以外*/
            printf("n の値が%d でしたので終了します\n",n);
            break; /* break は, ループを一つ抜ける*/
        }else{
            printf("n の値が 0 なので, 繰り返し処理を実行し続けます\n");
        }
    }
    return 0;
}
```

ポイント 4: 論理式は "0" または "1" の値を演算結果としている .

[例]

```
printf( " 0 == 1 の演算結果は%d である\n", 0==1);  
/* 0 < 1, 0 >1 の演算結果はどうなるのだろうか */
```

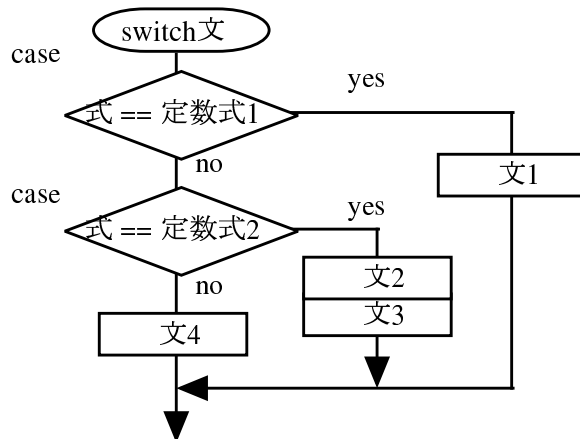
ポイント 5: “break” と同様にブロック内の処理を中断するが、繰り返しは中断しない場合 “continue” 文を用いる。while 文、do while 文では、条件式へ戻る。for 文では再設定の実行に移る。

example4\_3.c

```
/*  
 * continue 文の例  
 */  
# include <stdio.h>  
int main(void)  
{  
  int i, n, num;  
  printf("10 個の整数値を受け取り、正の整数を受け取った回数を示します。 \n");  
  for(i=0, num=0 ; i<10; i++){  
    printf("何か整数値を入力して下さい。 \n");  
    scanf("%d",&n);  
    if( n < 0){  
      continue;  
    }  
    num++;  
  }  
  printf("正の整数値は%d 回入力されました。 \n",num);  
  return 0;  
}
```

ポイント 6: 複数の分岐先の中から、条件に合うものを選ぶときには、switch ~ case 文を用いる

```
switch( 条件 ){  
  case 定数式 1 : 文 1;  
                 break;  
  case 定数式 2 : 文 2;  
                 文 3;  
                 break;  
  default:      文 4;  
}
```



example4\_4.c

```
/*  
 * continue 文の例  
 */  
# include <stdio.h>  
int main(void)  
{  
  int i, n, num;  
  printf("10 個の整数値を受け取り、正の整数を受け取った回数を示します。 \n");  
  for(i=0, num=0 ; i<10; i++){  
    printf("何か整数値を入力して下さい。 \n");  
    scanf("%d",&n);  
    if( n < 0){  
      continue;  
    }  
    num++;  
  }  
  printf("正の整数値は%d 回入力されました。 \n",num);  
  return 0;  
}
```