

# Cプログラミング入門

(応物2クラス)

第7回 繰り返し②

## 第7回講義の準備

---

- ▶ Course N@viから**第7回講義資料**をダウンロード
- ▶ **端末**を起動, ディレクトリ~/Cpro/07に移動
  - ▶ **cd Cpro/07**
- ▶ 共有ディレクトリ (**share**) のプログラムをコピー
  - ▶ **cp /share/dice10\_sum.c ./**

# 本日の講義・演習項目

---

- ▶ 繰り返し（復習）
- ▶ 繰り返しのテクニック
  - ▶ 便利な演算式
  - ▶ break文
  - ▶ 二重（多重）ループ
- ▶ 演習問題 ... 自主学習用です。レポートではありません。

# dice10\_sum.c

- ▶ サイコロを10回振ったときの目の合計値を出力するプログラム **dice10\_sum.c**

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i, num, sum = 0;
    for(i = 0 ; i < 10 ; i = i+1){
        num = rand( )%6+1;
        printf("Roll dice %d... %d¥n", i, num);
        sum = sum + num;
    }
    printf("Sum is %d¥n", sum);

    return 0;
}
```

前回講義で作成した  
プログラムと同じもの

# 繰り返しの復習

---

1. `dice10_sum.c` を **while文** を使って書き換えよ。
  - ▶ プログラム名 : **dice10\_while1.c**

# 復習 (for)

## for ・ while

### ▶ 基本構文 (for)

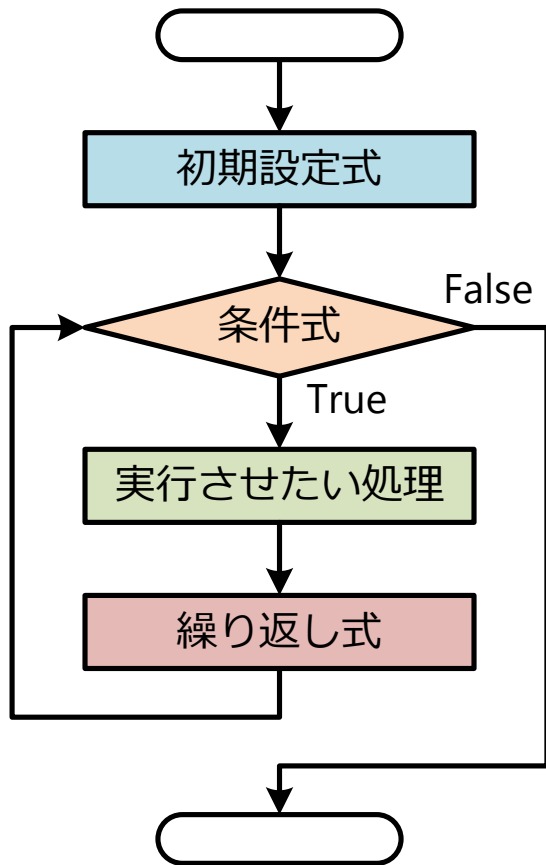
```
for( 初期設定式 ; 条件式 ; 繰り返し式 ){  
    繰り返し実行させたい処理  
}
```

- ▶ 初期設定式と繰り返し式には通常、**代入式**を記述
- ▶ 条件式には**比較演算子**を含む式を記述

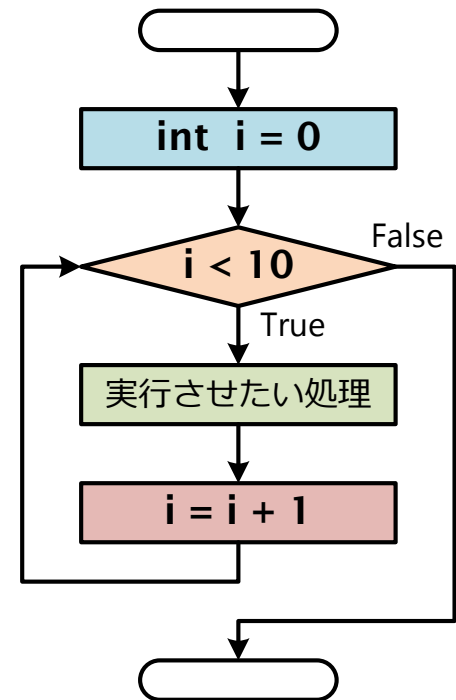
# 処理のイメージ

## ▶ for

```
for( 初期設定式 ; 条件式 ; 繰り返し式 ){  
    繰り返し実行させたい処理  
}
```



**10回** 繰り返し



# 復習 (while)

## for ・ while

### ▶ 基本構文 (while)

初期設定式

```
while( 条件式 ){  
    繰り返し実行させたい処理  
    繰り返し式  
}
```

- ▶ 初期設定式と繰り返し式には通常、**代入式**を記述
- ▶ 条件式には**比較演算子**を含む式を記述

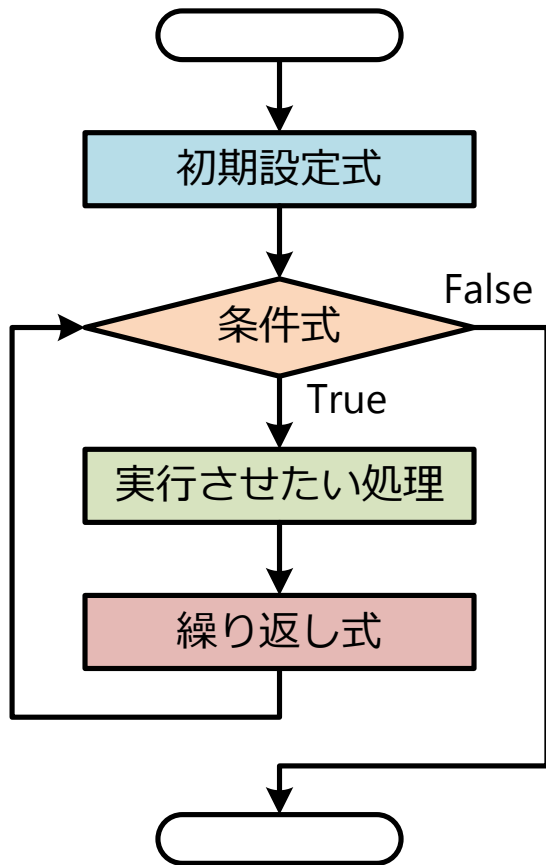


# 処理のイメージ

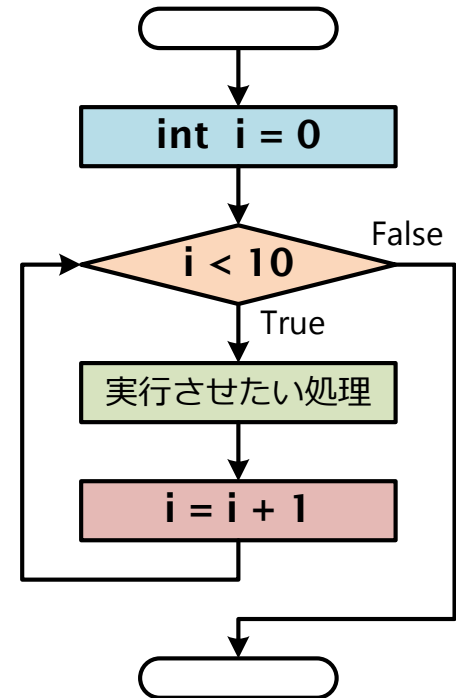
## ▶ while

初期設定式

```
while( 条件式 ){  
    繰り返し実行させたい処理  
    繰り返し式  
}
```



10回 繰り返し



# dice10\_while1.c

## プログラム例

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i = 0, num, sum = 0;
    while(i < 10){
        num = rand()%6+1;
        printf("Roll dice %d... %d\n", i, num);
        sum = sum + num;
        i = i + 1;
    }
    printf("Sum is %d\n", sum);

    return 0;
}
```

~:--- dice10\_while1.c All L18 (C/l Abbrev)

## 実行例

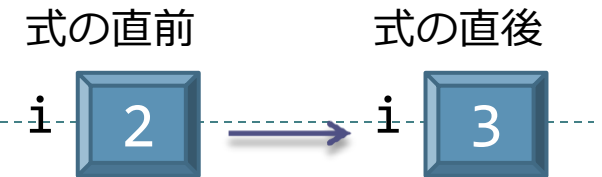
```
w536074@W63C037A:~/Cpro/07 - □ ×
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc dice10_while1.c -o dice10_while1
[w536074@W63C037A 07]$ ./dice10_while1
Roll dice 0... 6
Roll dice 1... 4
Roll dice 2... 4
Roll dice 3... 5
Roll dice 4... 6
Roll dice 5... 4
Roll dice 6... 3
Roll dice 7... 4
Roll dice 8... 6
Roll dice 9... 4
Sum is 46
[w536074@W63C037A 07]$
```

# 本日の講義・演習項目

---

- ▶ 繰り返し（復習）
- ▶ **繰り返しのテクニック**
  - ▶ 便利な演算式
  - ▶ break文
  - ▶ 二重（多重）ループ
- ▶ 演習問題

# 便利な演算式



演算式	式の意味
<code>i++</code>	<code>i = i + 1</code>
<code>i--</code>	<code>i = i - 1</code>
<code>i += a</code>	<code>i = i + a</code>
<code>i -= a</code>	<code>i = i - a</code>
<code>i *= a</code>	<code>i = i * a</code>
<code>i /= a</code>	<code>i = i / a</code>

- ▶ `i++` をインクリメント, `i--` をデクリメントと呼ぶ

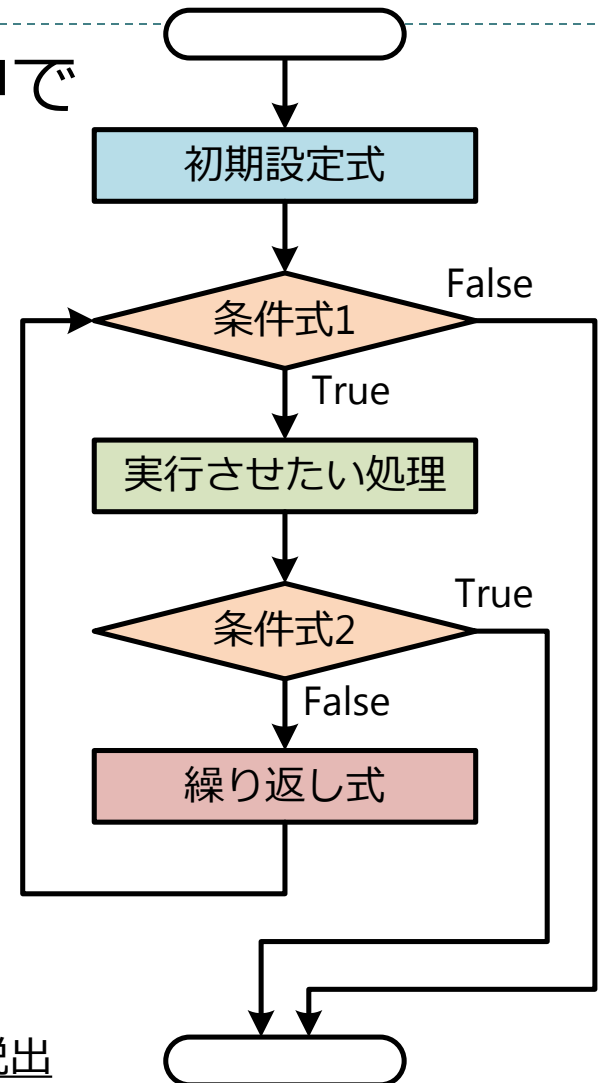
# break文

- ▶ 条件分岐によって、ループの**途中で**ループを脱出
- ▶ 記述例（while文）

## 初期設定式

```
while( 条件式1 ){  
    繰り返し実行させたい処理  
    if( 条件式2 ){  
        break;  
    }  
    繰り返し式  
}
```

※ 条件式1を満たす限り実行，条件式2を満たしたら脱出



## 二重（多重）ループ

---

- ▶ ループの内部でループを実行
- ▶ 記述例（for）

```
for( 初期設定式1 ; 条件式1 ; 繰り返し式1 ){  
    for( 初期設定式2 ; 条件式2 ; 繰り返し式2 ){  
        繰り返し実行させたい処理  
    }  
}
```

- ▶ 外側が **$N$ 回**，内側が **$M$ 回**ループする場合，計 **$N \times M$ 回**実行させたい処理が実行される
- ▶ 三重ループや四重ループにすることも可能

# 演習①

- ▶ dice10\_sum.c を便利な演算式を使って書き換えよ。
- ▶ プログラム名 : **dice10\_sum2.c**

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i, num, sum = 0;
    for(i = 0 ; i < 10 ; 
    }
    printf("Sum is %d¥n", sum);

    return 0;
}
```

便利な演算式を記述  
(動作は変わらない)

# dice10\_sum2.c

## プログラム例

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i, num, sum = 0;
    for(i = 0 ; i < 10 ; i++){
        num = rand()%6+1;
        printf("Roll dice %d... %d\n", i, num);
        sum += num;
    }
    printf("Sum is %d\n", sum);

    return 0;
}
```

~:\*\*\*- dice10\_sum2.c All L17 (C/l Abbrev)

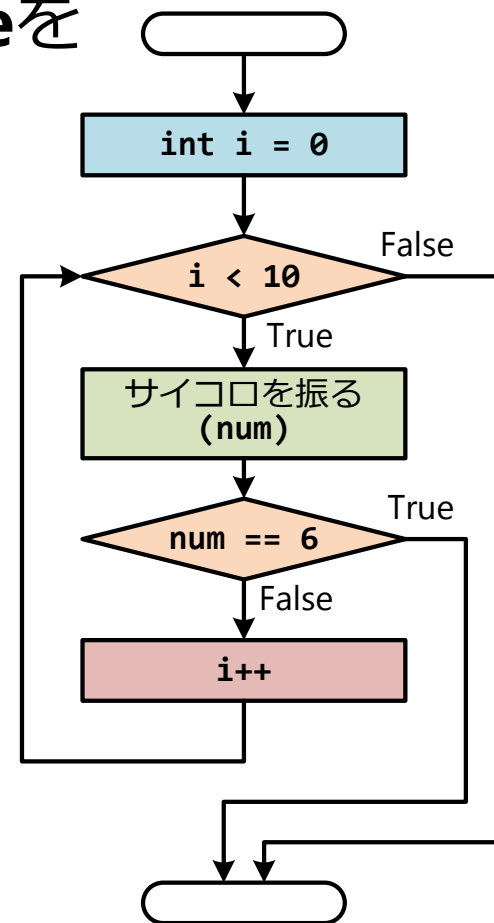
## 実行例

```
w536074@W63C037A:~/Cpro/07 - □ ×
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc dice10_sum2.c -o dice10_sum2
[w536074@W63C037A 07]$ ./dice10_sum2
Roll dice 0... 1
Roll dice 1... 4
Roll dice 2... 5
Roll dice 3... 5
Roll dice 4... 6
Roll dice 5... 1
Roll dice 6... 4
Roll dice 7... 5
Roll dice 8... 1
Roll dice 9... 4
Sum is 36
[w536074@W63C037A 07]$ █
```





## 演習②

- ▶ サイコロを10回振って1回でも6の目が出れば**win**を, 1回も出なければ**lose**を表示するプログラムを作成
  - ▶ プログラム名 : **dice10\_game.c**
- ▶ break文を使う
  - ▶ 右のフローチャートを参考にする
- ▶ ループを抜けた時点で,
  - ▶  **$i < 10 \rightarrow \text{win}$**
  - ▶  **$i == 10 \rightarrow \text{lose}$**



# dice10\_game.c (for文の場合)

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i, num;
    for(i = 0 ; i < 10 ; i++){
        num = rand( )%6+1;
        printf("Roll dice %d... %d¥n", i, num);
        
        }
        if() {
            printf("You win!!¥n");
        } else {
            printf("You lose.¥n");
        }
    }
    return 0;
}
```

} 6の目が出たらループ  
から脱出

} 勝ち負けの表示

# dice10\_game.c (while文の場合)

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i = 0, num;
    while(i < 10){
        num = rand( )%6+1;
        printf("Roll dice %d... %d¥n", i, num);
        ?
        i++;
    }
    if(?){
        printf("You win!!¥n");
    } else{
        printf("You lose.¥n");
    }
    return 0;
}
```

} 6の目が出たらループから脱出

} 勝ち負けの表示

# dice10\_game.c (for文の場合)

## プログラム例

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i, num;
    for(i = 0 ; i < 10 ; i++){
        num = rand()%6+1;
        printf("Roll dice %d... %d\n", i, num);
        if(num == 6){
            break;
        }
    }
    if(i < 10){
        printf("You win!!\n");
    } else{
        printf("You lose.\n");
    }

    return 0;
}
```

~:--- dice10\_game\_f.c All L23 (C/l Abbrev)

## 実行例

```
w536074@W63C037A:~/Cpro/07 - □ ×
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc dice10_game_f.c -o dice10_game
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 6
You win!!
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 1
Roll dice 1... 5
Roll dice 2... 5
Roll dice 3... 5
Roll dice 4... 5
Roll dice 5... 1
Roll dice 6... 2
Roll dice 7... 5
Roll dice 8... 4
Roll dice 9... 3
You lose.
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 3
Roll dice 1... 6
You win!!
[w536074@W63C037A 07]$ █
```

# dice10\_game.c (while文の場合)

## プログラム例

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(void){
    srand((unsigned)time(NULL));

    int i = 0, num;
    while(i < 10){
        num = rand()%6+1;
        printf("Roll dice %d... %d\n", i, num);
        if(num == 6){
            break;
        }
        i++;
    }
    if(i < 10){
        printf("You win!!\n");
    } else{
        printf("You lose.\n");
    }

    return 0;
}
```

~:\*\*\*- dice10\_game\_w.c All L24 (C/l Abbrev)

## 実行例

```
w536074@W63C037A:~/Cpro/07 - □ ×
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc dice10_game_f.c -o dice10_game
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 6
You win!!
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 1
Roll dice 1... 5
Roll dice 2... 5
Roll dice 3... 5
Roll dice 4... 5
Roll dice 5... 1
Roll dice 6... 2
Roll dice 7... 5
Roll dice 8... 4
Roll dice 9... 3
You lose.
[w536074@W63C037A 07]$ ./dice10_game
Roll dice 0... 3
Roll dice 1... 6
You win!!
[w536074@W63C037A 07]$ █
```

## 演習③

- ▶ 二重ループを使った**正方形**表示プログラム **square.c** を作成, 実行

```
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < len ; j++){
            printf("*");
        }
        printf("¥n");
    }

    return 0;
}
```

二重ループ

# square.c

```
square.c - emacs@W63C037A
File Edit Options Buffers Tools C Help

#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < len ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

-:--- square.c All L16 (C/l Abbrev)

## 実行例

```
w536074@W63C037A:~/Cpro/07
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc square.c -o square
[w536074@W63C037A 07]$ ./square
Input a natural number: 5
*****
*****
*****
*****
*****
[w536074@W63C037A 07]$ ./square
Input a natural number: 10
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
[w536074@W63C037A 07]$
```

## 演習④

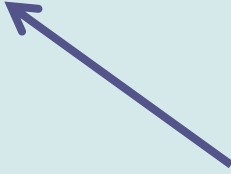
- ▶ 二重ループを使った**二等辺三角形**表示プログラム **triangle1.c** を作成, 実行

```
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j <= i ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

square.c からの変更点





# square.c

```
triangle1.c - emacs@W63C037A
File Edit Options Buffers Tools C Help

#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j <= i ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}

-:--- triangle1.c All L16 (C/l Abbrev)
```

## 実行例

```
w536074@W63C037A:~/Cpro/07
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc triangle1.c -o triangle1
[w536074@W63C037A 07]$ ./triangle1
Input a natural number: 5
*
**
***
****
*****
[w536074@W63C037A 07]$ ./triangle1
Input a natural number: 10
*
**
***
****
*****
*****
*****
*****
*****
*****
[w536074@W63C037A 07]$
```

# キーワード，次回の講義

---

- ▶ 本日のキーワード：
- ▶ 次回は6/7
- ▶ 次回講義までに予習ビデオ「第8回 配列・マクロ」を視聴し，各自プログラミング実習
- ▶ 前回講義で課題①を出題済み（×切：6/3）

# 本日の講義・演習項目

---

- ▶ 繰り返し（復習）
- ▶ 繰り返しのテクニック
  - ▶ 便利な演算式
  - ▶ break文
  - ▶ 二重（多重）ループ
- ▶ **演習問題**

# 演習問題 1 .


---



- ▶ 端末から入力した2以上の整数が**素数**であるかを判定するプログラムを作成せよ。
  - ▶ プログラム名 : prime1.c
- ▶ 素数とは, **1とその数以外で割り切れない数**のこと
- ▶ ヒント : プログラムのイメージは演習②に近い

# 演習問題 1 .

## ▶ プログラムの一部 (prime1.c)

```
#include <stdio.h>
int main(void){
    int i, num;
    printf("Input a natural number: ");
    scanf("%d", &num);

    for(i = 2 ; i < num ; i++){
        
    }

    if() {
        printf("not Prime\n");
    } else {
        printf("Prime\n");
    }
     判定結果の表示

    return 0;
}
```

# prime1.c

## プログラム例

```
#include <stdio.h>
int main(void){
    int i, num;
    printf("Input a natural number: ");
    scanf("%d", &num);

    for(i = 2 ; i < num ; i++){
        if(num % i == 0){
            break;
        }
    }
    if(i < num){
        printf("not Prime\n");
    } else{
        printf("Prime\n");
    }

    return 0;
}
```

--- prime1.c All L20 (C/l Abbrev)

## 実行例

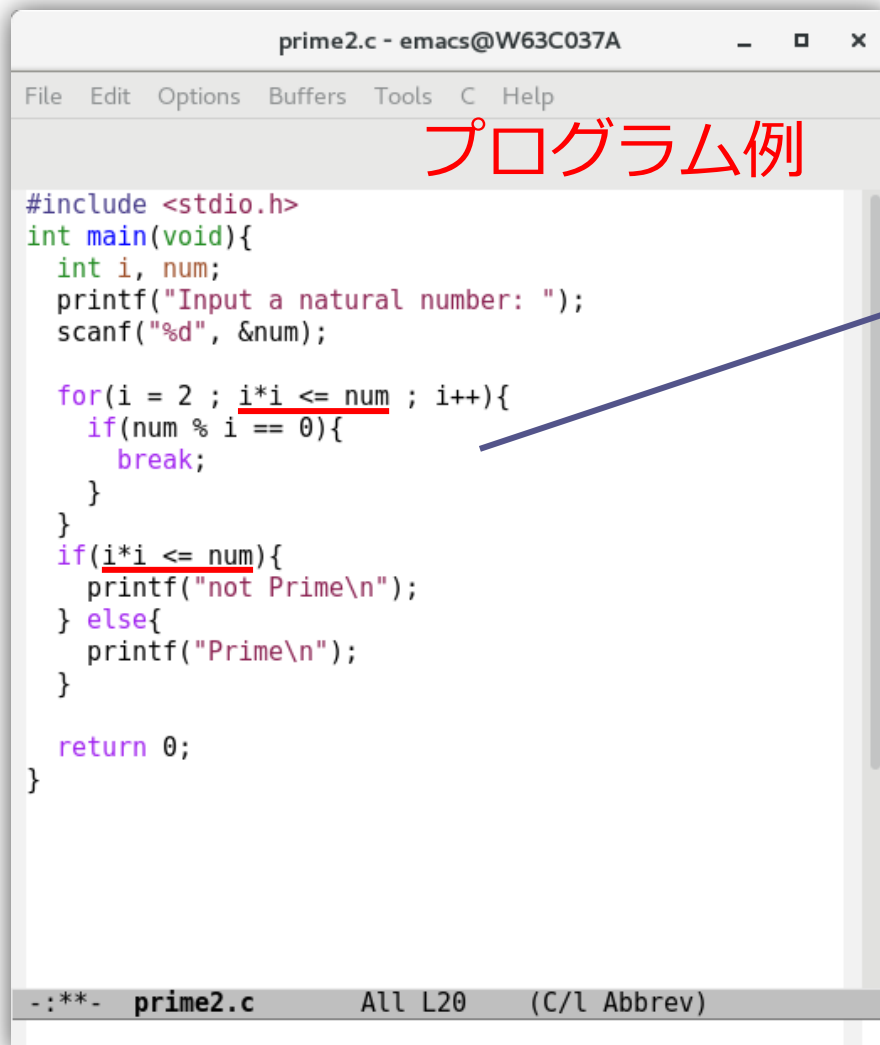
```
w536074@W63C037A:~/Cpro/07
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ gcc prime1.c -o prime1
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 2
Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 3
Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 4
not Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 5
Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 6
not Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 91
not Prime
[w536074@W63C037A 07]$ ./prime1
Input a natural number: 1047483637
Prime
[w536074@W63C037A 07]$
```

## 演習問題 2.

---

- ▶ 演習問題 1. の実行時に, **1047483637**を入力して判定結果を確認せよ。
  - ▶ 正しいプログラムであれば, Primeと表示されるはず
  - ▶ ただし, **実行に時間がかかる...** (2秒程度)
- ▶ 素数判定プログラムの**高速化**を検討し, 改良されたプログラム prime2.c を作成せよ。
  - ▶ 現状のプログラムにおいて無駄な処理はどこか?
  - ▶ 改良されたプログラムは, 上の判定を1秒未満で実行

# prime2.c



```
#include <stdio.h>
int main(void){
    int i, num;
    printf("Input a natural number: ");
    scanf("%d", &num);

    for(i = 2 ; i*i <= num ; i++){
        if(num % i == 0){
            break;
        }
    }
    if(i*i <= num){
        printf("not Prime\n");
    } else{
        printf("Prime\n");
    }

    return 0;
}
```

## プログラム例

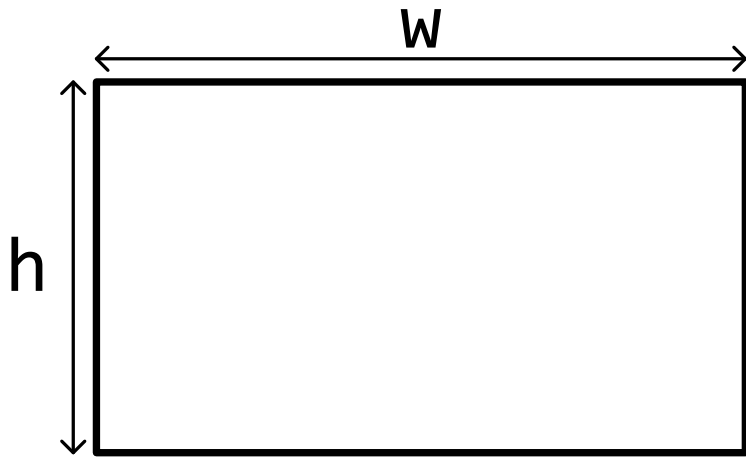
ある数 $n$ の約数のいずれか1つは  
 $\sqrt{n}$ 以下となる  
→  $i \times i$ が $n$ 以下となるような $i$ まで  
調査すれば十分



## 演習問題 3 .

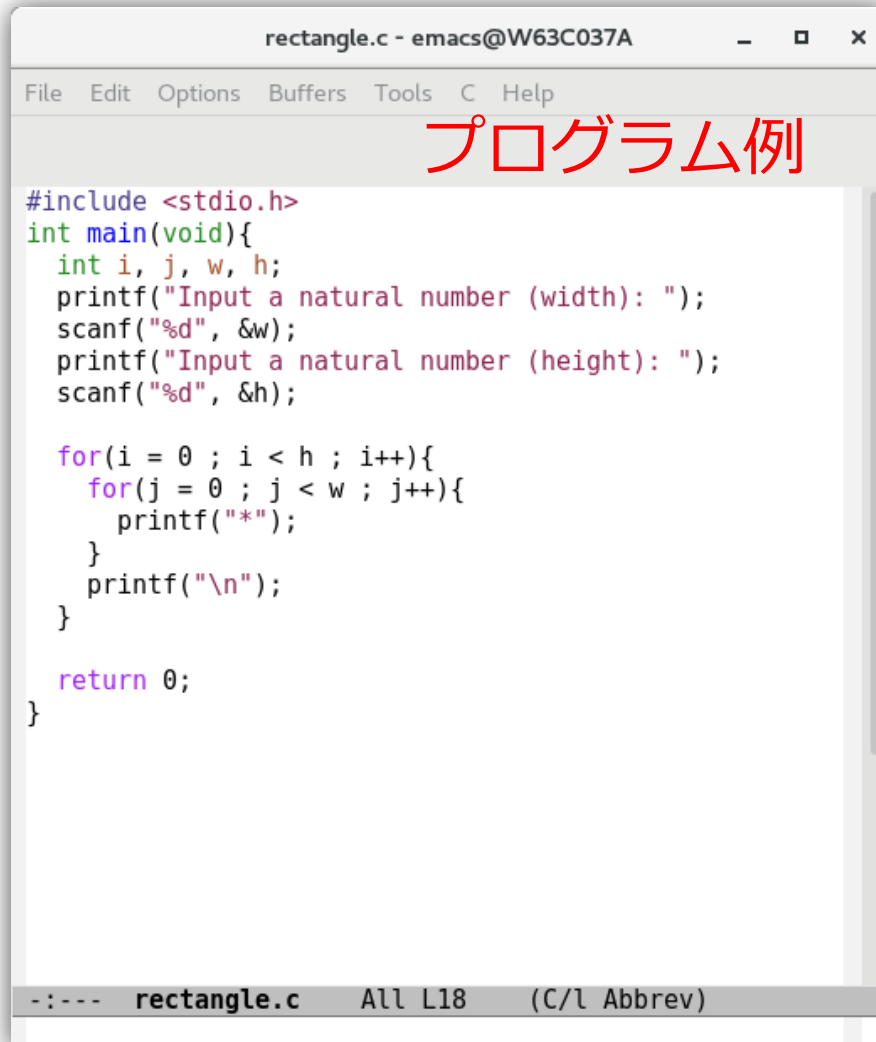
---

- ▶ 以下の図形を表示させるプログラムを作成せよ。
- ▶ ただし,  $w$ と $h$ は端末から入力できるようにすること。



rectangle.c

# rectangle.c



```
rectangle.c - emacs@W63C037A
File Edit Options Buffers Tools C Help
プログラム例
#include <stdio.h>
int main(void){
    int i, j, w, h;
    printf("Input a natural number (width): ");
    scanf("%d", &w);
    printf("Input a natural number (height): ");
    scanf("%d", &h);

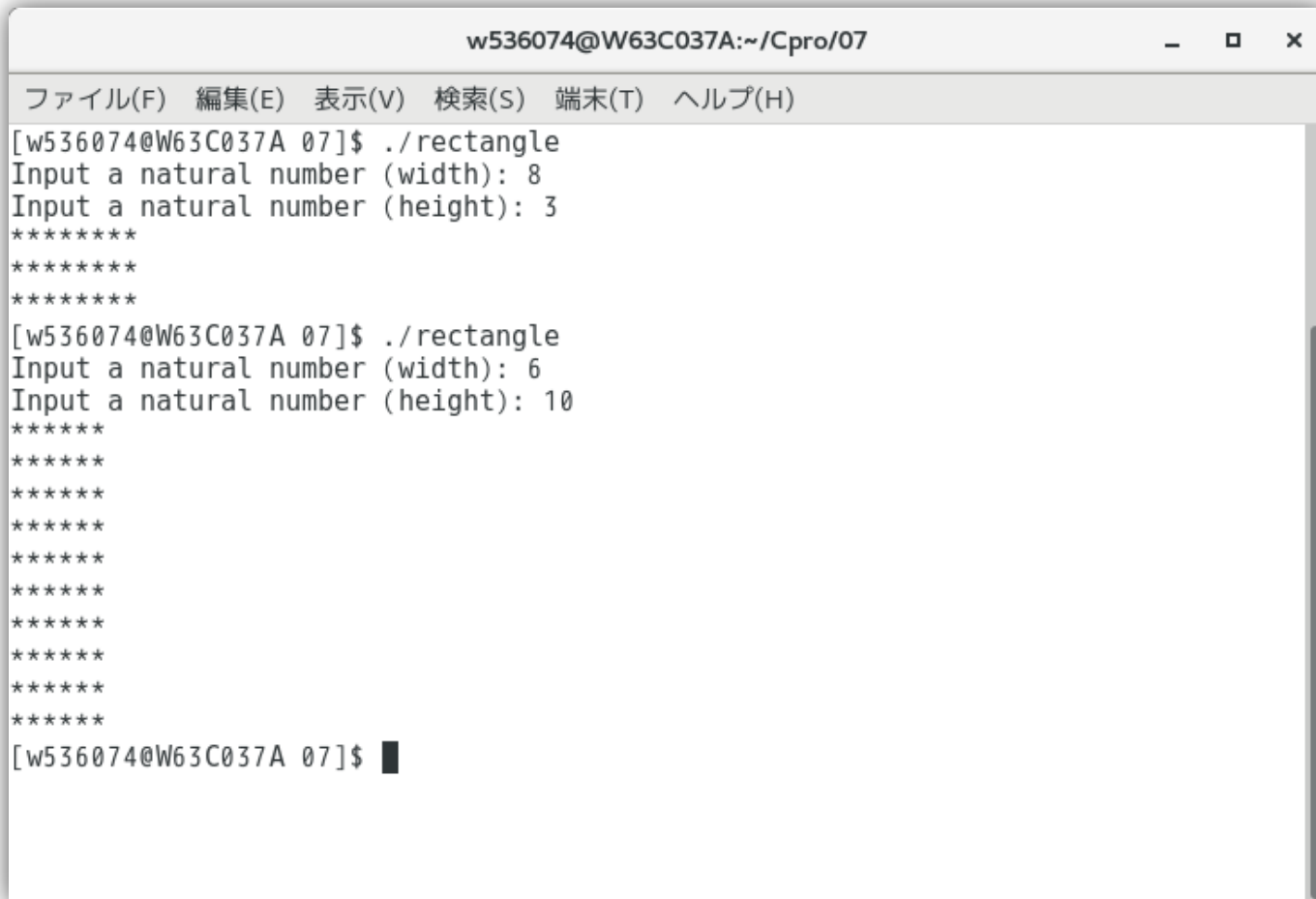
    for(i = 0 ; i < h ; i++){
        for(j = 0 ; j < w ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}

-:--- rectangle.c All L18 (C/l Abbrev)
```

# 演習問題 3 .

## ▶ 実行例

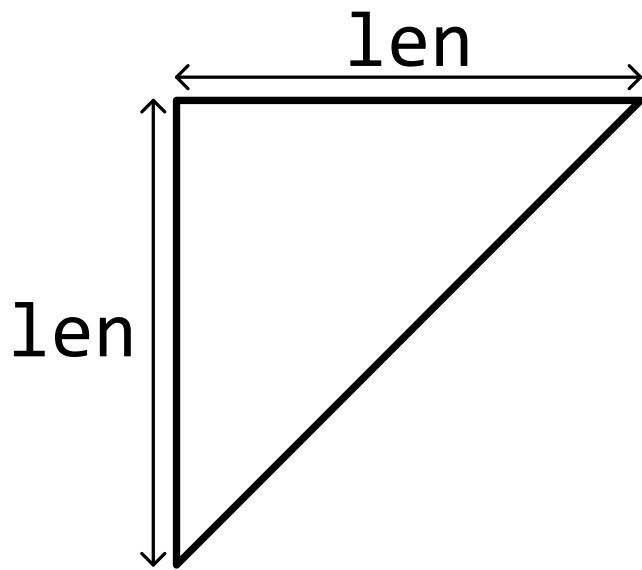


```
w536074@W63C037A:~/Cpro/07
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ ./rectangle
Input a natural number (width): 8
Input a natural number (height): 3
*****
*****
*****
[w536074@W63C037A 07]$ ./rectangle
Input a natural number (width): 6
Input a natural number (height): 10
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
[w536074@W63C037A 07]$
```

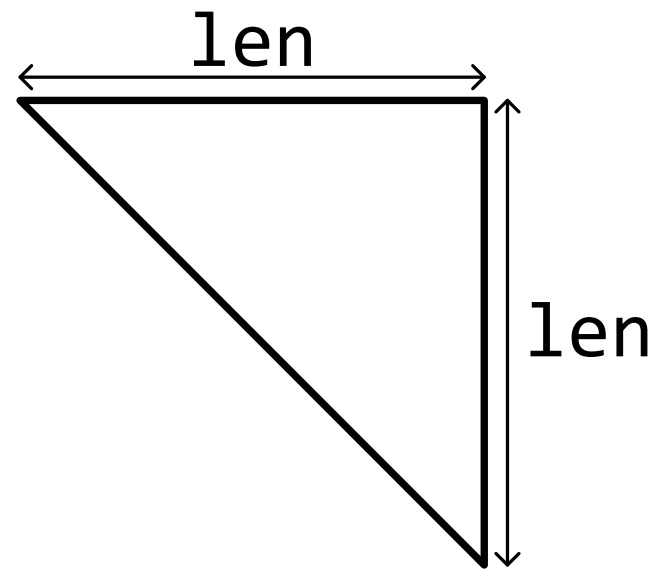
## 演習問題 4 .

---

- ▶ 以下の図形を表示させるプログラムを作成せよ。
- ▶ ただし, `len`は端末から入力できるようにすること。

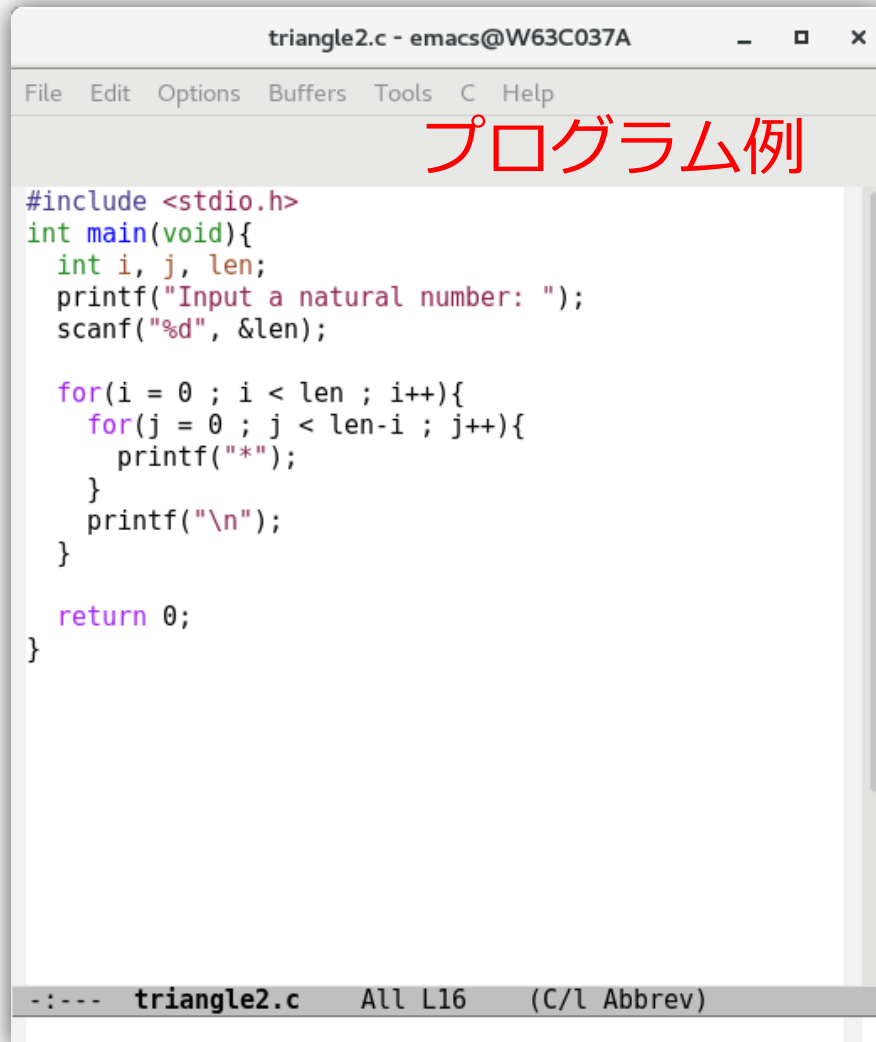


`triangle2.c`



`triangle3.c`

# triangle2.c



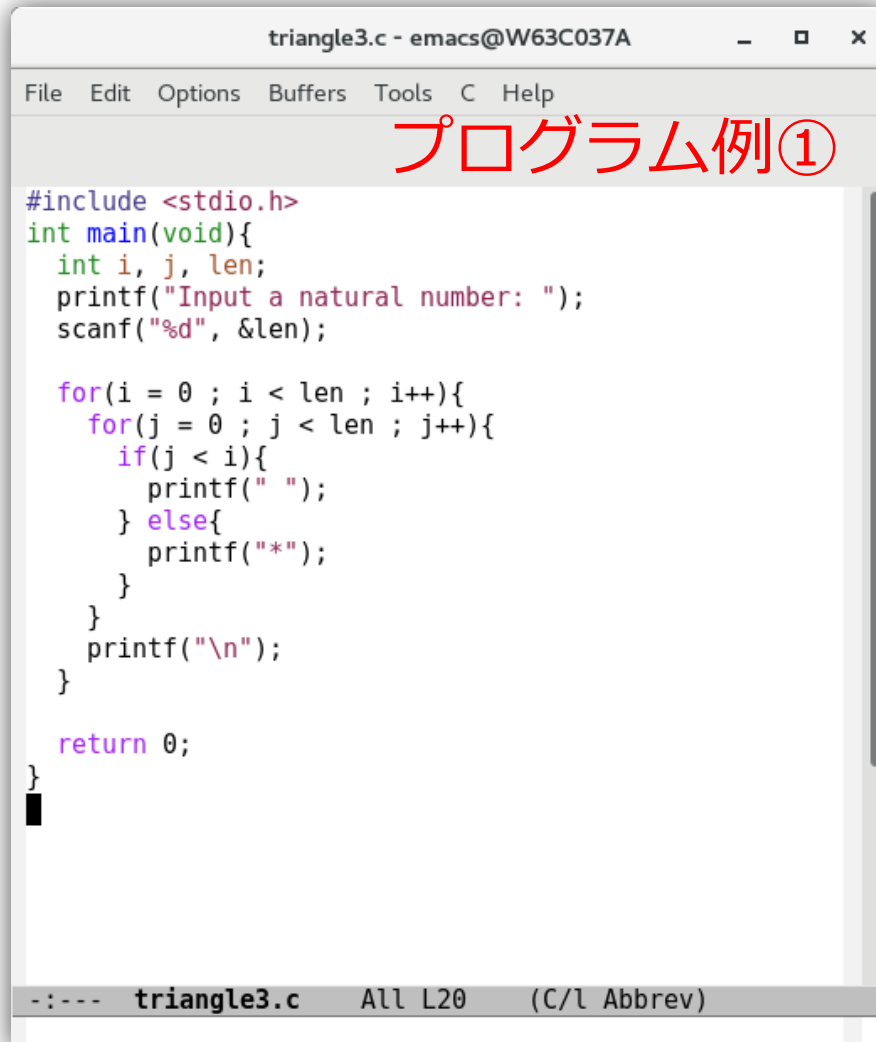
```
triangle2.c - emacs@W63C037A
File Edit Options Buffers Tools C Help
プログラム例
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < len-i ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}

-:--- triangle2.c All L16 (C/l Abbrev)
```

# triangle3.c



```
triangle3.c - emacs@W63C037A
File Edit Options Buffers Tools C Help
プログラム例①
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < len ; j++){
            if(j < i){
                printf(" ");
            } else{
                printf("*");
            }
        }
        printf("\n");
    }

    return 0;
}

-:--- triangle3.c All L20 (C/l Abbrev)
```

# triangle3.c (別解)

triangle3-1.c - emacs@W63C037A

File Edit Options Buffers Tools C Help

プログラム例②

```
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < i ; j++){
            printf(" ");
        }
        for(j = 0 ; j < len-i ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

--:--- triangle3-1.c All L19 (C/l Abbrev)

triangle3-2.c - emacs@W63C037A

File Edit Options Buffers Tools C Help

プログラム例③

```
#include <stdio.h>
int main(void){
    int i, j, len;
    printf("Input a natural number: ");
    scanf("%d", &len);

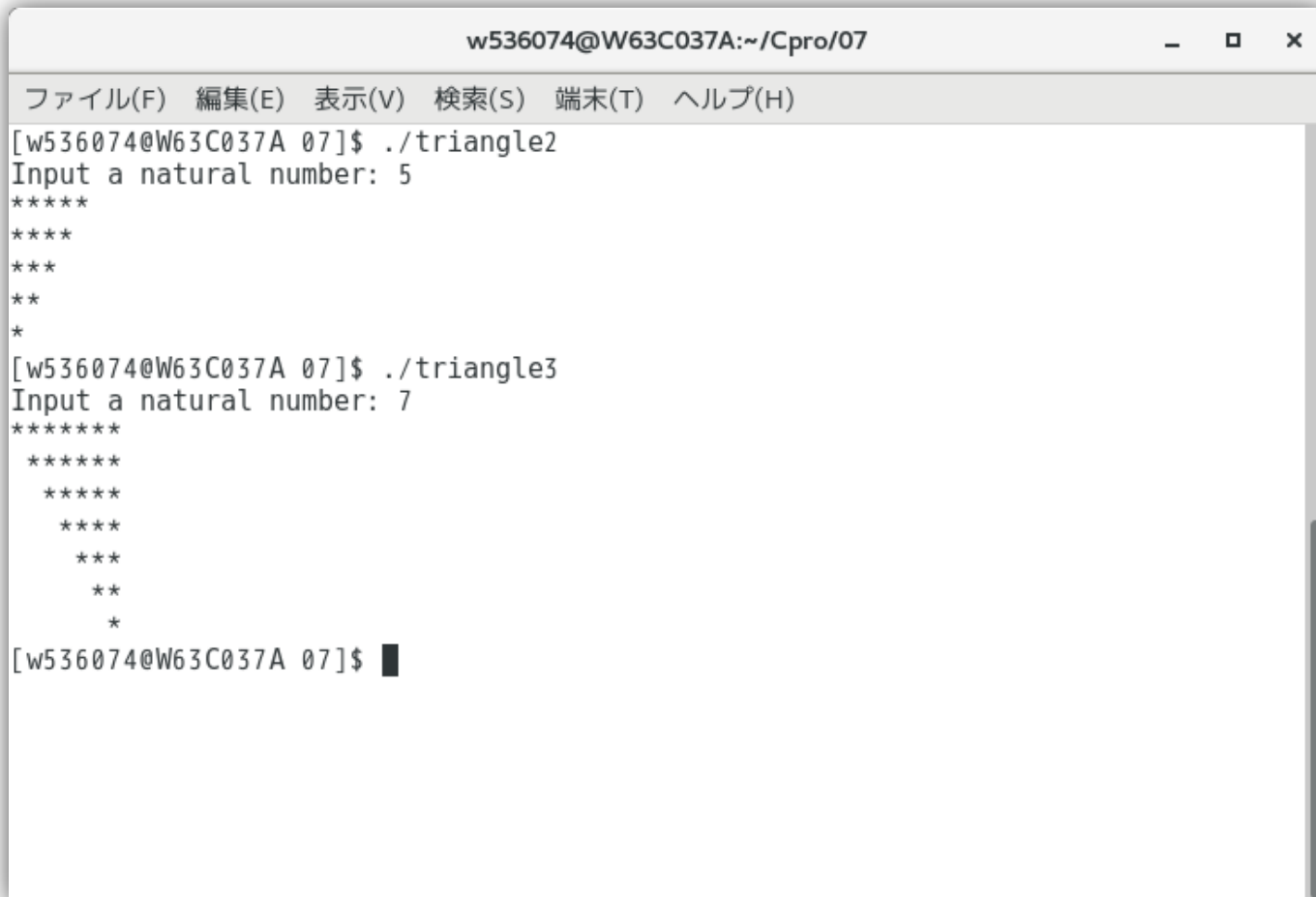
    for(i = 0 ; i < len ; i++){
        for(j = 0 ; j < i ; j++){
            printf(" ");
        }
        for( ; j < len ; j++){
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

--:--- triangle3-2.c All L19 (C/l Abbrev)

# 演習問題 4 .

## ▶ 実行例



```
w536074@W63C037A:~/Cpro/07
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[w536074@W63C037A 07]$ ./triangle2
Input a natural number: 5
*****
****
***
**
*
[w536074@W63C037A 07]$ ./triangle3
Input a natural number: 7
*****
*****
*****
****
***
**
*
[w536074@W63C037A 07]$
```