

プロ演1::課題3::説明資料1

演習トップページ

→ 問3 → 説明資料1 → START ボタン

<http://www.ced.is.utsunomiya-u.ac.jp>

[/lecture/2004/prog/p1/kadai3/lesson1/](http://www.ced.is.utsunomiya-u.ac.jp/lecture/2004/prog/p1/kadai3/lesson1/)

1. 複合代入演算子

代入式を簡略化する演算子。例：

```
a = a + b
```



```
a += b
```

合計10個：

```
+= -- *= /= %= <<= >>= &= ^= |=
```

2. 演算子の優先順位

優先順位(課題1参照)問題： $n = 1$ の場合

$$n+++2+1<<1*-1+2*+1$$

=

?

経験(教訓)：これだけ覚えればよい!

1. $+ - * /$ は、四則演算のとおり。

2. その他について、括弧をつけよう。

3. 練習問題(9分)

$n = 1$ とする。式

$$n+++2+1<<1*-1+2*+1$$

の値を考えよう。括弧をつけるなら
どうつけるかについて述べよう。
それをプログラムで確かめよう。

4. マクロ #define

前処理時に処理される定義のこと。例：

ASCIIコードを出力するプログラム

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

```
#define MAX 128
```

```
int main(void) {
```

```
    int i;
```

```
    for(i=0; i<MAX; i++)
```

```
        printf("%c", i);
```

```
}
```

5. #define の注意点

1. ;で終る必要がない。文字列に適用しない。
2. **そのまま**(展開して)置き換えられる。例

```
#define TWO 1+1
```

```
int i, j;
```

```
i = TWO; // i = 1+1; となる。
```

```
j = 2*TWO; // j = 2*1+1; となる。
```

正しい書き方：定義部に括弧をつける

6. 引数付きマクロ

引数付きのマクロ。例

絶対値を返すマクロ

```
#define ABS(x) ((x)<0?-(x):(x))
```

```
int n;
```

```
double f;
```

```
n = ABS(1); // n = ((1)<0?-(1):(1))
```

```
f = ABS(-1.2); // 結局 f が 1.2 に。
```

※ 関数みたいだが、展開されることに注意。

7. 練習問題(9分)

引数二つの最大値を返すマクロを書け。

プログラムで確かめよう。

余裕の人：最小値を返すマクロを考えよう。

引数二つの最大値を返すマクロ

```
#define MAX(x, y)
```

?

8. 関数の再帰呼び出し

関数が自分自身を呼び出すこと。例：

最大公約数を求める関数

```
/* 引数：正整数 a, b */  
int gcd(int a, int b) {  
    if (b == 0) return a;  
    return gcd(b, a%b);  
}
```

ユークリッドの互除法

9. 再帰呼び出しの注意点

正しく終了することの確認。

以下の例は再帰呼び出しではあるが...

終了条件を設けていない再帰呼び出し

```
/* 引数：正整数 a, b */
```

```
int wrong_gcd(int a, int b) {  
    return wrong_gcd(b, a%b);  
}
```

これを使うとどんな結果になる？

10. 練習問題(18分)

a. `wrong_gcd()`を使うとどんな結果になるかについて考えよう。確かめよう。

b. 再帰呼び出しを使って最大公約数を求めるプログラムを作成せよ。但し、標準入力から二つの正整数を読み込んで、計算した最大公約数を標準出力へ出力する。

11. 練習問題(19分)

c. 標準入力から正整数 n を読み込んで、 n 番目のフィボナッチ数を標準出力へ出力するプログラムを作成せよ。

フィボナッチ数列: 1 1 2 3 5 8 13 ...

$$f(1) = 1; f(2) = 1;$$

$$f(n) = f(n-1) + f(n-2) \quad n \geq 3$$

12. ☆ プログラムによる考察

m と n 番目のフィボナッチ数の最大公約数が
 m と n の最大公約数番目のフィボナッチ数に
等しいことを(何個かの m と n で)確かめよう。

例えば：12番目のフィボナッチ数は144，
8番目は21，144と21の最大公約数は3。
一方，12と8の最大公約数は4，4番目の
フィボナッチ数は3。等しい！