

DCP1203 HW6

10/24

Outline

- Announcements
- Lab quiz 1
- Class quiz 2
- Problems of HW6

Announcements

Announcements

- No class this week (make-up class on 11/4)

Lab Quiz 1

Problem 3

Enter the number of rows in pyramid of stars you wish to see 12

[illegible]

```
Program ended with exit code: 0
```

Problem 3

```
temp = n; // temp records the number of " " in each row
for (row = 1; row <= n; row++) {
    for (c = 1; c < temp; c++) printf(" ");

    temp--;

    for (c = 1; c <= (2 * row - 1); c++) printf("*");

    printf("\n");
}
```

Class Quiz 2

Operator precedence

- table
- Use parentheses ()

```
if (-1)
```

- Only 0 will be read as false

HW6: Problems

1

- Write a `function bool isPrime(unsigned int i)` that determines whether a number is prime. Use this function in a program that prints all prime factors of a positive integer input by the user. The maximum number we enter is 65535.
- One can use `bool` type in C by including `#include <stdbool.h>`

1

- Output

```
Enter a positive integer: 2772  
2772 has 4 prime factors: 2, 3, 7, 11
```

2

- Write a **recursive function** **power(base, exponent)** that, when invoked, returns $\text{base}^{\text{exponent}}$
- For example, `power(3, 4) = 3 * 3 * 3 * 3`. Assume that exponent is an integer greater than or equal to 1.

2

- Hint: The recursion step would use the relationship

$$\text{base}^{\text{exponent}} = \text{base} \cdot \text{base}^{(\text{exponent} - 1)}$$

- The terminating condition occurs when exponent is equal to 1 , because
$$\text{base}^1 = \text{base}$$

2

- Output

```
Enter base and exponent: 3 4  
The power(3, 4) is: 81
```


3

- Write a **recursive function gcd** that returns the greatest common divisor of **x** and **y**, defined recursively as follows: If **y** is equal to **0**, then **gcd(x, y)** is **x**; otherwise, **gcd(x, y)** is **gcd(y, x % y)**. [Note: For this algorithm, **x** must be larger than **y**.]

```
Please enter 2 numbers: 6 3
Their greatest common divisor is 3.
請按任意鍵繼續 . . .
```

4

- Write a **function lcm** that returns the **lowest common multiple** of **x** and **y**.

```
Enter 2 numbers: 24 12
```

```
Their lowest common multiple is 24.
```

```
請按任意鍵繼續 . . .
```

5

- Write a function that displays a solid square of asterisks (*) whose side is specified in integer parameter **side**.

Please enter the length of side: 4

```
* * * *  
* * * *  
* * * *  
* * * *
```

請按任意鍵繼續 . . .

6

- Please complete the **recursive function f** to calculate the n^{th} Fibonacci number.

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

int f(int n);

int main(){
    int n;

    printf("Enter a non-negative integer n (0~20): ");
    scanf("%d", n);
    printf("The nth Fibonacci number is %d.\n", f(n));

    system("PAUSE");
    return 0;
}

int f(int n){
    //Please finish the code here
    //請把這部分完成
}
```

Enter a non-negative integer n (0~20): 6
The nth Fibonacci number is 8.
請按任意鍵繼續 . . .