Flow of Control
Week 2
2017 Fall

Computer Programming for Engineers
Problem 1
My little calculator
My little calculator

- Put the formula to calculate
  - Formula format: [operand1] [operator] [operand2]
  - Operand type: integer
  - Operator: + - * / %

- Print the result of input formula
  - Use `switch` statement rather than `if-else`
  - Print "OP" instead for unexpected operands
  - Print "D0" instead for ‘Divide by 0’ condition

- Repeat the above for the result is non-zero
  - Terminate if the result is zero
My little calculator

1+2  3/0
3   D0
5-3 3%0
2   D0
3*2 3#3
6   0P
7/2 1*0
3   0
7%2
1
My little calculator

```cpp
#include <iostream>
using namespace std;

int main(){
    char op;
    int a,b;

    int res = 1;

    for(;;){
        cin >> a;
        cin >> op;
        cin >> b;

        switch (op) {
            case '+':
                res = a+b;
                break;
            case '-':
                res = a-b;
                break;
            case '*':
                res = a*b;
                break;
            ...
        }
    }
}
```

6-7: use appropriate datatype for ‘cin’

9: declare variable to store the result (Initialization is to prevent unexpected behavior, because there are branches that do not assign values to res below.)

11: use ‘for(;;)’ to start infinite loop using ‘for(; res; )’ is also nice

12-14: read inputs with ‘cin’ (Using ‘cin >> a >> op >> b;’ is not recommended, because blanks between inputs are needed, and the readability goes poor.)

16-25: use switch-case to store the appropriate result according to the operator
My little calculator

```cpp
case '/':
    if(!b){
        cout << "D0" << endl;
        continue;
    }
    res = a/b;
    break;

case '%':
    if(!b){
        cout << "D0" << endl;
        continue;
    }
    res = a%b;
    break;

default:
    cout << "OP" << endl;
    continue;
}

cout << res << endl;
if(!res)
    break;
```
Problem 2
Patterns of stars
Patterns of stars

• Put an integer, N
• Print 3 patterns of stars as follows
  • Triangle ( \_
  • Diagonal ( \_ )
  • Cross ( X )
• The sizes of the patterns are same as N
• Right-side blanks are arbitrary
Patterns of stars

> 3
*  
**  
***  

*  
*  
*  

* *  
*  
* *  

> 4
*  
**  
***  
****  

*  
*  
*  

* *  
* *  
* *  
* *
Patterns of stars

```cpp
#include <iostream>
using namespace std;

int main()
{
    int N;
    cin >> N;

    for(int i=1; i<=N; i++)
    {
        for(int j=0; j<i; j++)
        {
            cout << '*';
        }
        cout << endl;
    }
    cout << endl;
}
```

8: read input with ‘cin’

10: iterate printing i\textsuperscript{th} line. (i=1\ldots N)

11-14: print stars of \( i \) as the \( i \)\textsuperscript{th} line
Patterns of stars

18: iterate printing $i$th line. ($i=0\ldots N-1$)

19-22: print spaces of $i-1$ and a star as the $i$th line

26-27: iterate printing $(i, j)$ position
28-33: print star if the position is at diagonal ($i=j$), or space otherwise

※ ‘Diagonal’ is much easier to print using ‘if’.