**★☆☆☆☆**

**題組：Problem Set Archive** **with Online Judge**

**題號：10323: Factorial! You Must Be Kidding!**

**解題者：曾新惟**

**解題日期：2025年2月27日**

**題意：**

給定一個整數 n ，計算其階乘 n!。

(1). 在 [10000, 6227020800] 內，輸出 n! 的值

(2). 小於 10000 輸出 “Underflow!”

(3). 大於 6227020800 輸出 “Overflow!”

階乘公式 : F(n) = n \* F(n - 1)， F(0) = 1

(可變形) 例 : F(n) = n \* (n-1) \* F(n - 2)

**題意範例：**

2 🡺 Underflow!
10 🡺 3628800
100 🡺 Overflow!

**解法：**

1. 需先考慮 n 為負數的情況 !

公式移項: F(n) = n \* F(n - 1) 🡺 F(n-1) = F(n) / n

 因此 F(-1) = F(0) / 0 = 1 / 0 = + ∞ 🡺 Overflow!

 F(-2) = F(-1) / -1 = - ∞ 🡺 Underflow!

 (1) n < 0, n % 2 == 1 : Overflow!

 (2) n < 0, n % 2 == 0 : Underflow!

2. 判斷 n! 的可行範圍

 7! < 10000 < 8! < 13! = 6227020800 < 14!

 (1) 0 < n < 8 : Underflow!

 (2) 13 < n : Overflow!

 (3) else : n!

**解法範例：**

-10 🡺 n < 0, n % 2 == 0 🡺 Underflow!
-5 🡺 n < 0, n % 2 == 1 🡺 Overflow!
 5 🡺 0 < n < 8 🡺 Underflow!
10 🡺 8 < n < 13 🡺 3628800
15 🡺 13 < n 🡺 Overflow!

**討論：**

(1) 注意題目陷阱(特殊測資)

(2) 13! = 6227020800 超出 int 最大值，需使用long long

(3) 可以建立一個陣列儲存已經計算過的階乘值

(4) 因本題 n 範圍較小，可直接查表輸出結果

**程式：**

#include<iostream>

using namespace std;

int main(void){

 int n;

 while(cin>>n){

 if(n<0){

 if(n%2)

 cout<<"Overflow!"<<endl;

 else

 cout<<"Underflow!"<<endl;

 }

 else if (n<8)

 cout<<"Underflow!"<<endl;

 else if (n>13)

 cout<<"Overflow!"<<endl;

 else{

 long long x=1;

 for(int i=1;i<=n;i++)

 x\*=i;

 cout<<x<<endl;

 }

 }

 return 0;

}