CPP Problem Design Example

Subject: CPU bit growth

Contributor: 王聖文, 張子樂, 林岳儒

Main testing concept: 較大數字運算

Basics

■ C++ BASICS
■ FLOW OF CONTROL
□ FUNCTION BASICS
□ PARAMETERS AND OVERLOADING
□ ARRAYS
□ STRUCTURES AND CLASSES
□ CONSTRUCTORS AND OTHER TOOLS
□ OPERATOR OVERLOADING, FRIENDS,AND REFERENCES
□ STRINGS
□ POINTERS AND DYNAMIC ARRAYS

Functions

□ SEPARATE COMPILATION AND NAMESPACES
□ STREAMS AND FILE I/O
□ RECURSION
□ INHERITANCE
□ POLYMORPHISM AND VIRTUAL FUNCTIONS
□ TEMPLATES
□ LINKED DATA STRUCTURES
□ EXCEPTION HANDLING
□ STANDARD TEMPLATE LIBRARY
□ PATTERNS AND UML

Description:

Suppose a CPU with a k-bit can compute a maximum integer of (2^k) - 1, and every 10 years k will grow by a multiple of 2. Suppose that your company first released a 4-bit CPU in 1900, and the largest integer of its operation is 15 (so 8bits will be released in 1910, and 1920 is 16 bits... and so on).

Now given the year Y, find a maximum positive integer N, so that N! is within the CPU calculation range of the current year.

Test time limit: 5.0 seconds

Input:

Each line has a positive integer Y (2200 >= Y >= 1900).
The input method is unlimited input until the end of EOF is read.

Output:

Output N, so that N! is within the CPU calculation range of the current year.

Sample Input / Output：

Sample Input	Sample Output
1900	3
1910	5
2097	134480

□ Eazy,Only basic programming syntax and structure are required.
■ Medium,Multiple programming grammars and structures are required.
□ Hard,Need to use multiple program structures or complex data types.

Expected solving time:

25 minutes

Other notes: