CPP Problem Design Example

Subject: Number Game
Contributor: 陳泳峰, 袁瑋成, 范茗翔
Main testing concept: Class Implementation

Basics	Functions
■ C++ BASICS	☐ SEPARATE COMPILATION AND NAMESPACES
■ FLOW OF CONTROL	■ STREAMS AND FILE I/O
☐ FUNCTION BASICS	☐ RECURSION
■ PARAMETERS AND OVERLOADING	☐ INHERITANCE
☐ ARRAYS	☐ POLYMORPHISM AND VIRTUAL FUNCTIONS
■ STRUCTURES AND CLASSES	☐ TEMPLATES
■ CONSTRUCTORS AND OTHER TOOLS	☐ LINKED DATA STRUCTURES
■ OPERATOR OVERLOADING, FRIENDS, AND REFERENCES	☐ EXCEPTION HANDLING
☐ STRINGS	☐ STANDARD TEMPLATE LIBRARY
☐ POINTERS AND DYNAMIC ARRAYS	☐ PATTERNS AND UML

Description:

One day a mathematic major student designed a game about number but he/she have no idea how to implement it on computer. People major in computer science always love games and math, so he/she needs your help.

Here is the introduction of the game. Given a decimal A, and a number E. If E can be represented by the product of the digits of A, We will call the number E is valid. The game would give you a number list S. And our mission is find all these valid numbers in S.

E.g. Decimal A = 456 have three digits(a_1 = 4, a_2 = 5, a_3 = 6). When E = 30, E is a valid number because $a_2 * a_3$ = 5 * 6 = 30 = E. When E = 25 = 5 * 5, E is NOT a valid number because there is only one digit 5 in decimal A.

Remember our mission is to find all these valid numbers in S.

Create a class for this game. Here give you the definition of the class with some code comment as follows.

```cpp
////////////////////////////////////////////////////////////////
class NumberGame
{
private:
  int input;          // The decimal A.
  string fileName;    // That name of a file that contain the number list S.

  // You may add you own member variable.
public:
  void SetInput(int);               // set input
  void ProcessInput ();             // pre-process with your input.
  void SetFileName(string);         // set fileName
  void LoadNumberList();            // Read and pre-process the file
  void PrintAllValid();     // Print all the valid numbers in S ascendingly
  Void Reset()            // Reset every state/member.
  // You may implement you own member function.
};
```

Please implement the class according to the code comment.

Input:

The input is defined by the main function.

We will change the main function for testing.

There are at lease 5000000 numbers in S.

$1<=S[i]<= 2\text{\textasciicircum}32 - 1$

Output:

The output is defined by the main function.

We will change the main function for testing.

For each valid number in S just print once.

Sample Input / Output :

Sample Input	Sample Output

int main() {
   NumberGame Game;
   Game.SetInput(1234);
   Game.ProcessInput();
   Game.SetFileName("number.txt");
   Game.LoadNumberList();
   Game.PrintAllValid();
 	
   Game.Reset();
   cout << "\n";
   Game.SetInput(2345);
   Game.ProcessInput();
   Game.SetFileName("number.txt");
   Game.LoadNumberList();
   Game.PrintAllValid();
   system("pause");
}
	1
	2
	3
	4
	8
	24
	
	2
	3
	4
	5
	8
	24
	60

■ 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:

30 minutes

Other notes: