LintCode 参考程序

Lionx

2016 年 4 月 28 日

目录

第一章　入门（Naive）

1.1   Problem ID: 228 Middle of Linked List

1.1.1   Description

Find the middle node of a linked list.

1.1.2   Example

Given 1->2->3, return the node with value 2.

Given 1->2, return the node with value 1.

1.1.3   Code

C++

```cpp
/**
 * Definition of ListNode
 * class ListNode {
 * public:
 *     int val;
 *     ListNode *next;
 *     ListNode(int val) {
 *         this->val = val;
 *         this->next = NULL;
 *     }
 * }
 */
class Solution{
public:
    /**
     * @param head: the head of linked list.
     * @return: a middle node of the linked list
     */
    ListNode *middleNode(ListNode *head) {
        // Write your code here
        if (head == NULL){
            return NULL;
        }
        ListNode *fast = head;
        ListNode *slow = head;
        while(fast->next != NULL && fast->next->next != NULL){
            slow = slow->next;
            fast = fast->next->next;
        }
        return slow;
    }
};
```

Python

```python
"""
Definition of ListNode
class ListNode(object):

    def __init__(self, val, next=None):
        self.val = val
        self.next = next
"""

class Solution:
    # @param head: the head of linked list.
    # @return: a middle node of the linked list
    def middleNode(self, head):
        # Write your code here
        if head is None:
            return None
        slow = head;
        fast = head;
        while fast.next is not None and fast.next.next is not None:
            slow = slow.next
            fast = fast.next.next
        return slow
```

1.2   Problem ID: 366 Fibonacci

1.2.1   Description

Find the Nth number in Fibonacci sequence.

A Fibonacci sequence is defined as follow:

The first two numbers are 0 and 1.

The i th number is the sum of i-1 th number and i-2 th number.

The first ten numbers in Fibonacci sequence is:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34 ...

1.2.2   Example

Given 1, return 0

Given 2, return 1

Given 10, return 34

1.2.3   Code

C++

```cpp
class Solution{
public:
    /**
     * @param n: an integer
     * @return an integer f(n)
     */
    int fibonacci(int n) {
        // write your code here
        n--;
        double sqrt5=sqrt((double)5);
        return (pow((1+sqrt5),n)-pow((1-sqrt5),n))/(pow((double)2,n)*sqrt5);
    }
};
```

Python

```python
class Solution:
    # @param n: an integer
    # @return an integer f(n)
    def fibonacci(self, n):
        # write your code here
        a = 0;
```

```
7          b = 1;
8          for i in range(n - 1):
9              a, b = b, a + b
10         return a
```

1.3 Problem ID: 452 Remove Linked List Elements

1.3.1 Description

Remove all elements from a linked list of integers that have value val.

1.3.2 Example

Given 1->2->3->3->4->5->3, val = 3, you should return the list as 1->2->4->5

1.3.3 Code

C++

```cpp
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    /**
     * @param head a ListNode
     * @param val an integer
     * @return a ListNode
     */
    ListNode *removeElements(ListNode *head, int val) {
        // Write your code here
        ListNode dummy;
        dummy.next = head;
        head = &dummy;
        while (head->next != NULL) {
            if(head->next->val == val){
                head->next = head->next->next;
            }
            else{
            head = head->next;
            }
        }
        return dummy.next;
    }
};
```

Python

```python
# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, x):
#         self.val = x
#         self.next = None

class Solution:
    # @param head, a ListNode
    # @param val, an integer
    # @return a ListNode
    def removeElements(self, head, val):
        # Write your code here
        if head is None:
            return head
        dummy = ListNode(0)
        dummy.next = head
        head = dummy
        while head.next is not None:
            if head.next.val == val:
                head.next = head.next.next
            else:
                head = head.next
        return dummy.next
```

第二章　容易（Easy）

2.1   Problem ID: 1 A + B Problem

2.1.1   Description

Write a function that add two numbers A and B. You should not use + or any arithmetic operators.

2.1.2   Clarification

Are a and b both 32-bit integers?

Yes.

Can I use bit operation?

Sure you can.

2.1.3   Example

Given a=1 and b=2 return 3

2.1.4   Code

C++

```cpp
class Solution {
public:
    /*
     * @param a: The first integer
     * @param b: The second integer
     * @return: The sum of a and b
     */
    int aplusb(int a, int b) {
        // write your code here, try to do it without arithmetic operators.
        if(b == 0){
            return a;
        }
        else{
            aplusb(a ^ b, (a & b) << 1);
        }
    }
};
```

Python

```python
1  class Solution:
2      """
3      @param a: The first integer
4      @param b: The second integer
5      @return:  The sum of a and b
6      """
7
8      def aplusb(self, a, b):
9          # write your code here, try to do it without arithmetic operators.
10         return a + b
```

Note: Python version will be Updated in future.

2.2   Problem ID: 2 Trailing Zeros

2.2.1   Description

Write an algorithm which computes the number of trailing zeros in n factorial.

2.2.2   Example

11! = 39916800, so the out should be 2

2.2.3   Code

C++

```cpp
1  class Solution {
2   public:
3      // param n : description of n
4      // return: description of return
5      long long trailingZeros(long long n) {
6          long long sum = 0;
7          while(n!=0){
8              sum += n / 5;
9              n = n / 5;
10         }
11         return sum;
12     }
13 };
```

Python

```python
class Solution:
    # @param n a integer
    # @return ans a integer
    def trailingZeros(self, n):
        sum = 0
        while n != 0:
            sum += n // 5
            n = n // 5
        return sum
```

2.3   Problem ID: 6 Merge Sorted Array II

2.3.1   Description

Merge two given sorted integer array A and B into a new sorted integer array.

2.3.2   Example

A=[1,2,3,4]
B=[2,4,5,6]
return [1,2,2,3,4,4,5,6]

2.3.3   Code

C++

```cpp
class Solution {
public:
    /**
     * @param A and B: sorted integer array A and B.
     * @return: A new sorted integer array
     */
    vector<int> mergeSortedArray(vector<int> &A, vector<int> &B) {
        // write your code here
        vector<int> C;
        int i = 0, j = 0;
        while(i < A.size() && j < B.size()){
            if(A[i] < B[j]){
                C.push_back(A[i++]);
            }else{
                C.push_back(B[j++]);
```

```
16                }
17            }
18            while(i < A.size()){
19                C.push_back(A[i++]);
20            }
21            while(j < B.size()){
22                C.push_back(B[j++]);
23            }
24            return C;
25        }
26  };
```

Python

```python
class Solution:
    #@param A and B: sorted integer array A and B.
    #@return: A new sorted integer array
    def mergeSortedArray(self, A, B):
        # write your code here
        C = []
        i = 0
        j = 0
        while i < len(A) and j < len(B):
            if A[i] < B[j]:
                C.append(A[i])
                i = i + 1
            else:
                C.append(B[j])
                j = j + 1
        if(i < len(A)):
            C.extend(A[i:])
        if(j < len(B)):
            C.extend(B[j:])
        return C
```

2.4   Problem ID: 8 Rotate String

2.4.1   Description

Given a string and an offset, rotate string by offset. (rotate from left to right)

2.4.2   Example

Given "abcdefg".
offset=0 => "abcdefg"

offset=1 => "gabcdef"

offset=2 => "fgabcde"

offset=3 => "efgabcd"

2.4.3   Code

C++

```
1    class Solution {
2    public:
3        /**
4         * @param str: a string
5         * @param offset: an integer
6         * @return: nothing
7         */
8        void rotateString(string &str,int offset){
9            //wirte your code here
10           if (str.size() == 0){
11
12           }else{
13               offset = offset % str.size();
14               str = str.substr(str.size() − offset, offset) +
15               str.substr(0, str.size() − offset);
16           }
17       }
18   };
```

Python

```
1    class Solution:
2        # @param s: a list of char
3        # @param offset: an integer
4        # @return: nothing
5        def rotateString(self, s, offset):
6                # write you code here
7                if len(s) != 0:
8                    offset = offset % len(s)
9                    s[:] = s[−offset:] + s[:−offset]
```

2.5   Problem ID: 9 Fizz Buzz

2.5.1   Description

Given number n. Print number from 1 to n. But:

when number is divided by 3, print "fizz".

when number is divided by 5, print "buzz".

when number is divided by both 3 and 5, print "fizz buzz".

2.5.2   Example

If n = 15, you should return:

[

"1", "2", "fizz",

"4", "buzz", "fizz",

"7", "8", "fizz",

"buzz", "11", "fizz",

"13", "14", "fizz buzz"

]

2.5.3   Code

C++

```cpp
class Solution {
public:
    /**
     * param n: As description.
     * return: A list of strings.
     */
    vector<string> fizzBuzz(int n) {
        vector<string> results;
        for (int i = 1; i <= n; i++) {
            if (i % 15 == 0) {
                results.push_back("fizz buzz");
            } else if (i % 5 == 0) {
                results.push_back("buzz");
            } else if (i % 3 == 0) {
                results.push_back("fizz");
            } else {
                results.push_back(to_string(i));
            }
        }
        return results;
    }
};
```

Python

```python
1   class Solution:
2       """
3       @param n: An integer as description
4       @return: A list of strings.
5       For example, if n = 7, your code should return
6           ["1", "2", "fizz", "4", "buzz", "fizz", "7"]
7       """
8       def fizzBuzz(self, n):
9           results = []
10          for i in range(1, n+1):
11              if i % 15 == 0:
12                  results.append("fizz buzz")
13              elif i % 5 == 0:
14                  results.append("buzz")
15              elif i % 3 == 0:
16                  results.append("fizz")
17              else:
18                  results.append(str(i))
19          return results
```

2.6 Problem ID: 13 strStr

2.6.1 Description

For a given source string and a target string, you should output the first index(from 0) of target string in source string.

If target does not exist in source, just return -1.

2.6.2 Clarification

Do I need to implement KMP Algorithm in a real interview?

Not necessary. When you meet this problem in a real interview, the interviewer may just want to test your basic implementation ability. But make sure your confirm with the interviewer first.

2.6.3 Example

If source = "source" and target = "target", return -1.

If source = "abcdabcdefg" and target = "bcd", return 1.

2.6.4   Code

C++

```cpp
#include <cstring>
#include <iostream>
using namespace std;
class Solution {
public:
    /**
     * Returns a index to the first occurrence of target in source,
     * or -1  if target is not part of source.
     * @param source string to be scanned.
     * @param target string containing the sequence of characters to match.
     */
    int strStr(const char *source, const char *target) {
        // write your code here
        if (source == NULL || target == NULL){
            return -1;
        }
        int size_source = strlen(source);
        int size_target = strlen(target);
        int i, j;
        for (i = 0; i < size_source - size_target + 1; i++){
            for(j = 0; j <size_target; j++){
                if(source[i + j] != target[j]){
                    break;
                }
            }
            if(j == size_target){
                return i;
            }
        }
        return -1;
    }
};
```

Python

```python
class Solution:
    def strStr(self, source, target):
        # write your code here
        if source is None or target is None:
            return -1
        return source.find(target)
```

2.7 Problem ID: 14 First Position of Target

2.7.1 Description

For a given sorted array (ascending order) and a target number, find the first index of this number in O(log n) time complexity.

If the target number does not exist in the array, return -1.

2.7.2 Example

If the array is [1, 2, 3, 3, 4, 5, 10], for given target 3, return 2.

2.7.3 Code

C++

```cpp
#include <cstring>
#include <iostream>
using namespace std;
class Solution {
public:
    /**
     * Returns a index to the first occurrence of target in source,
     * or −1  if target is not part of source.
     * @param source string to be scanned.
     * @param target string containing the sequence of characters to match.
     */
    int strStr(const char *source, const char *target) {
        // write your code here
        if (source == NULL || target == NULL){
            return −1;
        }
        int size_source = strlen(source);
        int size_target = strlen(target);
        int i, j;
        for (i = 0; i < size_source − size_target + 1; i++){
            for(j = 0; j <size_target; j++){
                if(source[i + j] != target[j]){
                    break;
                }
            }
            if(j == size_target){
                return i;
            }
        }
        return −1;
    }
};
```

Python

```python
class Solution:
    # @param nums: The integer array
    # @param target: Target number to find
    # @return the first position of target in nums, position start from 0
    def binarySearch(self, nums, target):
        # write your code here
        start = 0
        end = len(nums) - 1
        while start + 1 < end:
            mid = (start + end) >> 1
            if nums[mid] < target:
                start = mid
            else:
                end = mid
        if nums[start] == target:
            return start
        if nums[end] == target:
            return end
        return -1
```

2.8   Problem ID: 28 Search a 2D Matrix

2.8.1   Description

Write an efficient algorithm that searches for a value in an m x n matrix.

This matrix has the following properties:

Integers in each row are sorted from left to right.

The first integer of each row is greater than the last integer of the previous row.

2.8.2   Example

Consider the following matrix:

[

[1, 3, 5, 7],

[10, 11, 16, 20],

[23, 30, 34, 50]

]

Given target = 3, return true.

2.8.3　Code

C++

```cpp
class Solution {
public:
    /**
     * @param matrix, a list of lists of integers
     * @param target, an integer
     * @return a boolean, indicate whether matrix contains target
     */
    bool searchMatrix(vector<vector<int> > &matrix, int target) {
        // write your code here
        int n = matrix.size();
        if (n == 0){
            return false;
        }
        int m = matrix[0].size();
        if (m == 0){
            return false;
        }

        int start = 0;
        int end = m * n - 1;
        int mid, row ,col;

        while (start + 1 < end){
            mid = start + (end - start) / 2;
            row = mid / m;
            col = mid % m;
            if (matrix[row][col] == target){
                return true;
            }else if(matrix[row][col] < target){
                start = mid;
            }else{
                end = mid;
            }
        }
        if (matrix[start / m][start % m] == target){
            return true;
        }
        if (matrix[end / m][end % m] == target){
            return true;
        }
        return false;
    }
};
```

Python

```python
class Solution:
    """
    @param matrix, a list of lists of integers
    @param target, an integer
    @return a boolean, indicate whether matrix contains target
    """
    def searchMatrix(self, matrix, target):
        # write your code here
        m = len(matrix)
        if m == 0:
            return False
        n = len(matrix[0])
        if n == 0:
            return False
        start = 0
        end = m * n - 1
        while start + 1 < end:
            mid = (start + end) / 2
            x = mid / n
            y = mid % n
            if matrix[x][y] < target:
                start = mid
            else:
                end = mid
        x, y = start / n, start % n
        if matrix[x][y] == target:
            return True
        x, y = end / n, end % n
        if matrix[x][y] == target:
            return True
        return False
```

2.9   Problem ID: 82 Single Number

2.9.1   Description

Given 2*n + 1 numbers, every numbers occurs twice except one, find it.

2.9.2   Example

Given [1,2,2,1,3,4,3], return 4

2.9.3　Code

C++

```cpp
class Solution {
public:
    /**
     * @param A: Array of integers.
     * return: The single number.
     */
    int singleNumber(vector<int> &A) {
        // write your code here
        int x;
        for(int i = 0; i < A.size(); i++){
            x ^= A[i];
        }
        return x;
    }
};
```

Python

```python
class Solution:
        """
        @param A : an integer array
        @return : a integer
        """
        def singleNumber(self, A):
                # write your code here
                x = 0
                for a in A:
                        x ^= a
                return x
```

2.10　Problem ID: 157 Unique Characters

2.10.1　Description

Implement an algorithm to determine if a string has all unique characters.

2.10.2　Example

Given "abc", return true.

Given "aab", return false.

2.10.3   Code

C++

```cpp
class Solution {
public:
    /**
     * @param str: a string
     * @return: a boolean
     */
    bool isUnique(string &str) {
        // write your code here
        int ch[128] = {0};
        for(int i = 0; i < str.length(); i++){
            if(ch[str[i]] != 0){
                return false;
            }else{
                ch[str[i]] = 1;
            }
        }
        return true;
    }
};
```

Python

```python
class Solution:
    # @param s: a string
    # @return: a boolean
    def isUnique(self, str):
        # write your code here
        ch = range(129)
        for i in range(129):
            ch[i] = 0
        for s in str:
            if ch[ord(s)] != 0:
                return False
            ch[ord(s)] = ch[ord(s)] + 1
        return True
```

2.11   Problem ID: 166 Nth to Last Node in List

2.11.1   Description

Find the nth to last element of a singly linked list.

The minimum number of nodes in list is n.

2.11.2   Example

Given a List 3->2->1->5->null and n = 2, return node whose value is
1.

2.11.3   Code

C++

```cpp
/**
 * Definition of ListNode
 * class ListNode {
 * public:
 *     int val;
 *     ListNode *next;
 *     ListNode(int val) {
 *         this->val = val;
 *         this->next = NULL;
 *     }
 * }
 */
class Solution {
public:
    /**
     * @param head: The first node of linked list.
     * @param n: An integer.
     * @return: Nth to last node of a singly linked list.
     */
    ListNode *nthToLast(ListNode *head, int n) {
        // write your code here
        if(head == NULL || n < 1){
            return NULL;
        }
        ListNode *p1 = head;
        ListNode *p2 = head;
        for(int i = 0; i < n; i++){
            if(p2 != NULL){
                p2 = p2->next;
            }else{
                return NULL;
            }
        }
        while(p2 != NULL){
            p1 = p1->next;
            p2 = p2->next;
        }
        return p1;
    }
```

```
40  };
```

Python

```python
1   """
2   Definition of ListNode
3   class ListNode(object):
4
5       def __init__(self, val, next=None):
6           self.val = val
7           self.next = next
8   """
9   class Solution:
10      """
11      @param head: The first node of linked list.
12      @param n: An integer.
13      @return: Nth to last node of a singly linked list.
14      """
15      def nthToLast(self, head, n):
16          # write your code here
17          if head is None or n < 1:
18              return None
19          p1 = head
20          p2 = head
21          for i in range(0,n):
22              if p2 is not None:
23                  p2 = p2.next
24              else:
25                  return None
26          while p2 is not None:
27              p2 = p2.next
28              p1 = p1.next
29          return p1
```

2.12 Problem ID: 496 Toy Factory

2.12.1 Description

Factory is a design pattern in common usage. Please implement a ToyFactory which can generate proper toy based on the given type.

2.12.2 Example

```
1   ToyFactory tf = ToyFactory();
2   Toy toy = tf.getToy('Dog');
3   toy.talk();
```

```
4  >> Wow
5
6  toy = tf.getToy('Cat');
7  toy.talk();
8  >> Meow
```

2.12.3   Code

C++

```cpp
1  /**
2   * Your object will be instantiated and called as such:
3   * ToyFactory* tf = new ToyFactory();
4   * Toy* toy = tf->getToy(type);
5   * toy->talk();
6   */
7  class Toy {
8  public:
9      virtual void talk() const=0;
10 };
11
12 class Dog: public Toy {
13     // Write your code here
14     void talk() const{
15         cout << "Wow" << endl;
16     }
17 };
18
19 class Cat: public Toy {
20     // Write your code here
21     void talk() const{
22         cout << "Meow" << endl;
23     }
24 };
25
26 class ToyFactory {
27 public:
28     /**
29      * @param type a string
30      * @return Get object of the type
31      */
32     Toy* getToy(string& type) {
33         // Write your code here
34         if(type == "Dog"){
35             return new Dog();
36         }
37         if(type == "Cat"){
38             return new Cat();
39         }
40         return NULL;
41     }
42 };
```

Python

```python
"""
Your object will be instantiated and called as such:
ty = ToyFactory()
toy = ty.getToy(type)
toy.talk()
"""
class Toy:
    def talk(self):
        raise NotImplementedError('This method should have implemented.')

class Dog(Toy):
    # Write your code here
    def talk(self):
        print "Wow"

class Cat(Toy):
    # Write your code here
    def talk(self):
        print "Meow"


class ToyFactory:
    # @param {string} shapeType a string
    # @return {Toy} Get object of the type
    def getToy(self, type):
        # Write your code here
        if type == "Dog":
            return Dog()
        if type == "Cat":
            return Cat()
        return None
```

2.13   Problem ID: 497 Shape Factory

2.13.1   Description

Factory is design pattern in common usage. Implement a ShapeFactory that can generate correct shape.

2.13.2   Example

```
ShapeFactory sf = new ShapeFactory();
```

```
2   Shape shape = sf.getShape("Square");
3   shape.draw();
4   >>  ——
5   >> |     |
6   >> |     |
7   >>  ——
8
9   shape = sf.getShape("Triangle");
10  shape.draw();
11  >>   /\
12  >>  /  \
13  >> /____\
14
15  shape = sf.getShape("Rectangle");
16  shape.draw();
17  >>  ——
18  >> |     |
19  >>  ——
```

2.13.3   Code

C++

```cpp
1   /**
2    * Your object will be instantiated and called as such:
3    * ShapeFactory* sf = new ShapeFactory();
4    * Shape* shape = sf->getShape(shapeType);
5    * shape->draw();
6    */
7   class Shape {
8   public:
9       virtual void draw() const=0;
10  };
11
12  class Rectangle: public Shape {
13      // Write your code here
14      void draw() const{
15          cout << "␣——" << endl << "|␣␣␣␣|" << endl << "␣——" << endl;
16      }
17  };
18
19  class Square: public Shape {
20      // Write your code here
21        void draw() const{
22          cout << "␣——" << endl << "|␣␣␣␣|" << endl
23          << "|␣␣␣␣|" << endl << "␣——" << endl;
24        }
25  };
26
27  class Triangle: public Shape {
28      // Write your code here
29        void draw() const{
```

```
30            cout << "␣␣/\\" << endl << "␣/␣␣\\\" << endl << "/____\\\" << endl;
31        }
32  };
33
34  class ShapeFactory {
35  public:
36      /**
37       * @param shapeType a string
38       * @return Get object of type Shape
39       */
40      Shape* getShape(string& shapeType) {
41          // Write your code here
42          if(shapeType == "Square"){
43              return new Square();
44          }
45          if(shapeType == "Rectangle"){
46              return new Rectangle();
47          }
48          if(shapeType == "Triangle"){
49              return new Triangle();
50          }
51          return NULL;
52      }
53  };
```

Python

```
1   """
2   Your object will be instantiated and called as such:
3   sf = ShapeFactory()
4   shape = sf.getShape(shapeType)
5   shape.draw()
6   """
7   class Shape:
8       def draw(self):
9           raise NotImplementedError('This␣method␣should␣have␣implemented.')
10
11  class Triangle(Shape):
12      # Write your code here.
13      def draw(self):
14          print "␣␣/\\"
15          print "␣/␣␣\\\"
16          print "/____\\\"
17
18  class Rectangle(Shape):
19      # Write your code here
20      def draw(self):
21          print "␣────"
22          print "|␣␣␣␣|"
23          print "␣────"
24
25  class Square(Shape):
```

```python
26          # Write your code here
27          def draw(self):
28              print "␣────"
29              print "|␣␣␣␣|"
30              print "|␣␣␣␣|"
31              print "␣────"
32
33  class ShapeFactory:
34      # @param {string} shapeType a string
35      # @return {Shape} Get object of type Shape
36      def getShape(self, shapeType):
37          # Write your code here
38          if shapeType == "Square":
39              return Square()
40          if shapeType == "Triangle":
41              return Triangle()
42          if shapeType == "Rectangle":
43              return Rectangle()
44          return None
```

第三章　中等（Medium）

3.1   Problem ID: 3 Digit Counts

3.1.1   Description

Count the number of k's between 0 and n. k can be 0 - 9.

3.1.2   Example

if n=12, k=1 in [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12], we have FIVE 1's (1, 10, 11, 12)

3.1.3   Code

C++

```cpp
class Solution {
public:
    /*
     * param k : As description.
     * param n : As description.
     * return: How many k's between 0 and n.
     */
    int digitCounts(int k, int n) {
        // write your code here
        int count = 0;
        for(int i = 0; i <= n; i++){
            int j = i;
            while(true){
                if(j % 10 == k){
                    count++;
                }
                j = j / 10;
                if(j == 0){
                    break;
                }
            }
        }
        return count;
    }
};
```

Python

```python
class Solution:
    # @param k & n  two integer
    # @return ans a integer
    def digitCounts(self, k, n):
```

```python
 5            assert(n >= 0 and 0 <= k <= 9)
 6            count = 0
 7            for i in range(n + 1):
 8                j = i
 9                while True:
10                    if j % 10 == k:
11                        count = count + 1
12                    j = j // 10
13                    if j == 0:
14                        break
15            return count
```

3.2   Problem ID: 4 Ugly Number II

3.2.1   Description

Ugly number is a number that only have factors 2, 3 and 5.

Design an algorithm to find the nth ugly number. The first 10 ugly numbers are 1, 2, 3, 4, 5, 6, 8, 9, 10, 12...

Notice Note that 1 is typically treated as an ugly number.

3.2.2   Example

If n=9, return 10.

3.2.3   Code

C++

```cpp
class Solution {
public:
    /*
     * @param n an integer
     * @return the nth prime number as description.
     */
    int nthUglyNumber(int n) {
        // write your code here
        int *uglys = new int[n];
        uglys[0] = 1;
        int next = 1;
        int *p2 = uglys;
        int *p3 = uglys;
        int *p5 = uglys;
        while(next < n){
            int m = min(min(*p2 * 2, *p3 * 3), *p5 *5);
            uglys[next] = m;
            while(*p2 * 2 <= uglys[next]){
                p2++;
            }
            while(*p3 * 3 <= uglys[next]){
                p3++;
            }
            while(*p5 * 5 <= uglys[next]){
                p5++;
            }
            next++;
        }
        int uglyNum = uglys[n - 1];
        delete[] uglys;
```

```
31          return uglyNum;
32      }
33 };
```

Python

```python
1  class Solution:
2      """
3      @param {int} n an integer.
4      @return {int} the nth prime number as description.
5      """
6      def nthUglyNumber(self, n):
7          # write your code here
8          uglys = []
9          uglys.append(1)
10         p2 = 0
11         p3 = 0
12         p5 = 0
13         next = 1
14         while next < n:
15             m = min(uglys[p2] * 2, uglys[p3] * 3, uglys[p5] * 5)
16             uglys.append(m)
17             while uglys[p2] * 2 <= uglys[next]:
18                 p2 = p2 + 1
19             while uglys[p3] * 3 <= uglys[next]:
20                 p3 = p3 + 1
21             while uglys[p5] * 5 <= uglys[next]:
22                 p5 = p5 + 1
23             next = next + 1
24         return uglys[n - 1]
```

3.3   Problem ID: 5 Kth Largest Element

3.3.1   Description

Find K-th largest element in an array.

3.3.2   Example

In array [9,3,2,4,8], the 3rd largest element is 4.

In array [1,2,3,4,5], the 1st largest element is 5, 2nd largest element is 4, 3rd largest element is 3 and etc.

3.3.3   Code

C++

```cpp
1   class Solution {
2   public:
3       /*
4        * param k : description of k
5        * param nums : description of array and index 0 ~ n−1
6        * return: description of return
7        */
8       int kthLargestElement(int k, vector<int> nums) {
9           // write your code here
10          sort(nums.rbegin(), nums.rend());
11          return nums[k−1];
12      }
13  };
```

Python

```python
1   class Solution:
2       # @param k & A a integer and an array
3       # @return ans a integer
4       def kthLargestElement(self, k, A):
5           A = sorted(A, reverse = True)
6           return A[k−1]
```

Note: This problem will be update soon!

3.4   Problem ID: 394 Coins in a Line

3.4.1   Description

There are n coins in a line. Two players take turns to take one or two coins from right side until there are no more coins left. The player who take the last coin wins.

Could you please decide the first play will win or lose?

3.4.2   Example

n = 1, return true.

n = 2, return true.

n = 3, return false.

n = 4, return true.

n = 5, return true.

3.4.3   Code

C++

```cpp
class Solution {
public:
    /**
     * @param n: an integer
     * @return: a boolean which equals to true if the first player will win
     */
    bool firstWillWin(int n) {
        // write your code here
        if(n % 3 == 0){
            return false;
        }
        return true;
    }
};
```

Python

```python
class Solution:
    # @param n: an integer
    # @return: a boolean which equals to True if the first player will win
    def firstWillWin(self, n):
        # write your code here
```

```
6        if n % 3 == 0:
7            return False
8        return True
```

3.5   Problem ID: 419 Roman to Integer

3.5.1   Description

Given a roman numeral, convert it to an integer.

The answer is guaranteed to be within the range from 1 to 3999.

3.5.2   Clarification

What is Roman Numeral?

https://en.wikipedia.org/wiki/Roman_numerals

https://zh.wikipedia.org/wiki/%E7%BD%97%E9%A9%AC%E6%95%B0%E5%AD%97

http://baike.baidu.com/view/42061.htm

3.5.3   Example

IV -> 4

XII -> 12

XXI -> 21

XCIX -> 99

3.5.4   Code

C++

```cpp
class Solution {
public:
    /**
     * @param s Roman representation
     * @return an integer
     */
    int romanToInt(string& s) {
        // Write your code here
        map<char, int> num;
        num['I'] = 1;
        num['V'] = 5;
        num['X'] = 10;
        num['L'] = 50;
        num['C'] = 100;
        num['D'] = 500;
        num['M'] = 1000;
        int ans = num[s[s.length() - 1]];
        for (int i = s.length() - 2; i >= 0; i--) {
```

```
19                if (num[s[i]] < num[s[i + 1]]){
20                    ans -= num[s[i]];
21                }
22                else{
23                    ans += num[s[i]];
24                }
25            }
26            return ans;
27        }
28    };
```

Python

```python
1    class Solution:
2        # @param {string} s Roman representation
3        # @return {int} an integer
4        def romanToInt(self, s):
5            # Write your code here
6            if s == "":
7                return 0
8            roman = {'I':1, 'V':5, 'X':10, 'L':50, 'C':100, 'D':500, 'M':1000}
9            sum = roman[s[-1]]
10           i = len(s) - 2
11           while i >= 0:
12               if roman[s[i]] < roman[s[i + 1]]:
13                   sum -= roman[s[i]]
14               else:
15                   sum += roman[s[i]]
16               i -= 1
17           return sum
```

3.6   Problem ID: 428 Pow(x, n)

3.6.1   Description

Implement pow(x, n).

3.6.2   Example

Pow(2.1, 3) = 9.261

Pow(0, 1) = 0

Pow(1, 0) = 1

3.6.3   Code

C++

```cpp
class Solution {
public:
    /**
     * @param x the base number
     * @param n the power number
     * @return the result
     */
    double myPow(double x, int n) {
        // Write your code here
        if(n < 0){
            return 1.0 / myPow(x, -n);
        }
        if(n == 0){
            return 1;
        }
        if(n % 2 == 0){
            return  myPow(x, n>>1) * myPow(x, n>>1);
        }else{
            return  myPow(x, n>>1) * myPow(x, n>>1) * x;
        }
    }
};
```

Python

```python
class Solution:
    # @param {double} x the base number
    # @param {int} n the power number
    # @return {double} the result
    def myPow(self, x, n):
```

```
 6              # Write your code here
 7              if n < 0:
 8                  return 1.0 / self.myPow(x, -n)
 9              if n == 0:
10                  return 1
11              if n % 2 == 0:
12                  return self.myPow(x, n >> 1) * self.myPow(x, n >> 1)
13              else:
14                  return self.myPow(x, n >> 1) * self.myPow(x, n >> 1) * x
```

第四章　困难（Hard）

第五章　超难（Super）