**算法训练 阶乘**

时间限制：1.0s   内存限制：512.0MB

**问题描述**  
　　一个整数n的阶乘可以写成n!，它表示从1到n这n个整数的乘积。阶乘的增长速度非常快，例如，13!就已经比较大了，已经无法存放在一个整型变量中；而35!就更大了，它已经无法存放在一个浮点型变量中。因此，当n比较大时，去计算n!是非常困难的。幸运的是，在本题中，我们的任务不是去计算n!，而是去计算n!最右边的那个非0的数字是多少。例如，5! = 1\*2\*3\*4\*5 = 120，因此5!最右边的那个非0的数字是2。再如：7! = 5040，因此7!最右边的那个非0的数字是4。请编写一个程序，输入一个整数n(n<=100)，然后输出n! 最右边的那个非0的数字是多少。  
　　输入格式：输入只有一个整数n。  
　　输出格式：输出只有一个整数，即n! 最右边的那个非0的数字。  
**输入输出样例**

样例输入

6

样例输出

2

本题的C++参考代码如下：

#include <iostream>

#include <cmath>

#include <string>

#include <windows.h>

#include <stack>

#include <vector>

using namespace std;

int main(int argc, char\*\* argv) {

vector<int> cs;

int n;

cin >> n;

cs.push\_back(1);

for (int i = 2; i <= n; i++)

{

for (int j = 0; j<cs.size(); j++)

{

cs[j] \*= i;

}

for (int j = 0; j<cs.size() - 1; j++)

{

if (cs[j]>9)

{

cs[j + 1] += cs[j] / 10;

cs[j] = cs[j] % 10;

}

}

while (cs[cs.size() - 1]>9)

{

cs.push\_back(cs[cs.size() - 1] / 10);

cs[cs.size() - 2] = cs[cs.size() - 2] % 10;

}

}

for (int i = 0; i < cs.size() ; i++)

{

if (cs[i]!=0)

{

cout << (char)(cs[i] + '0');

break;

}

}

return 0;

}

本题的C参考代码如下：

//常州轻工职业技术学院七七队长提供

#include<stdio.h>

int main(void)

{

int n;

int i;

int sum=1;

scanf("%d",&n);

for(i=n;i>=1;i--)

{

sum = sum\*i;

while(sum%10==0)

sum=sum/10;

if(sum/100>0)

sum=sum%100;

}

printf("%d",sum%10);

return 0;

}

本题的Java参考代码如下：

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.math.BigDecimal;

public class Main {

public static void main(String[] args) throws Exception {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int number = Integer.parseInt(br.readLine());

BigDecimal bd = new BigDecimal(1);

for (long i = 1; i <= number; i++) {

bd = bd.multiply(new BigDecimal(i));

}

String out = bd.toString();

for (int i = out.length() - 1; i >= 0; i--) {

if (out.charAt(i) != '0') {

System.out.println(out.charAt(i));

return;

}

}

}

}