**问题描述**

|  |  |
| --- | --- |
| 试题编号： | 201312-2 |
| 试题名称： | ISBN号码 |
| 时间限制： | 1.0s |
| 内存限制： | 256.0MB |
| 问题描述： | **问题描述**  　　每一本正式出版的图书都有一个ISBN号码与之对应，ISBN码包括9位数字、1位识别码和3位分隔符，其规定格式如“x-xxx-xxxxx-x”，其中符号“-”是分隔符（键盘上的减号），最后一位是识别码，例如0-670-82162-4就是一个标准的ISBN码。ISBN码的首位数字表示书籍的出版语言，例如0代表英语；第一个分隔符“-”之后的三位数字代表出版社，例如670代表维京出版社；第二个分隔之后的五位数字代表该书在出版社的编号；最后一位为识别码。 　　识别码的计算方法如下： 　　首位数字乘以1加上次位数字乘以2……以此类推，用所得的结果mod 11，所得的余数即为识别码，如果余数为10，则识别码为大写字母X。例如ISBN号码0-670-82162-4中的识别码4是这样得到的：对067082162这9个数字，从左至右，分别乘以1，2，…，9，再求和，即0×1+6×2+……+2×9=158，然后取158 mod 11的结果4作为识别码。 　　编写程序判断输入的ISBN号码中识别码是否正确，如果正确，则仅输出“Right”；如果错误，则输出是正确的ISBN号码。  **输入格式**  　　输入只有一行，是一个字符序列，表示一本书的ISBN号码（保证输入符合ISBN号码的格式要求）。  **输出格式**  　　输出一行，假如输入的ISBN号码的识别码正确，那么输出“Right”，否则，按照规定的格式，输出正确的ISBN号码（包括分隔符“-”）。  **样例输入**  0-670-82162-4  **样例输出**  Right  **样例输入**  0-670-82162-0  **样例输出**  0-670-82162-4 |
| 代码程序 | #include<stdio.h>  #include<string.h>  int main()  {  char ISBN[20];  char ISBNnumber[20];  scanf("%s", ISBN);  // 预处理  int len = strlen(ISBN);  int lenN = 0;  for(int i = 0; i< len-1;i++) // 提取数字(不包含最后一个)  {  if(ISBN[i]>='0' && ISBN[i]<='9')  {  ISBNnumber[lenN++] = ISBN[i];  }  }  // 计算  int result = 0;  for(int i = 0; i<lenN;i++)  {  result+=(ISBNnumber[i] - '0')\*(i+1);  }  result = result %11;  char Map[11] = {'0','1','2','3','4','5','6','7','8','9','X'};  if(Map[result] == ISBN[len-1])  {  printf("Right\n");  }else{  ISBN[len-1] = Map[result];  printf("%s\n",ISBN);  }  return 0;  } |