# **中国大学MOOC-翁恺-C语言程序设计习题集**

PAT 习题集

**02-0. 整数四则运算(10)**

本题要求编写程序，计算2个正整数的和、差、积、商并输出。题目保证输入和输出全部在整型范围内。

****输入格式：****

输入在一行中给出2个正整数A和B。

****输出格式：****

在4行中按照格式“A 运算符 B = 结果”顺序输出和、差、积、商。

****输入样例：****

3 2

****输出样例：****

3 + 2 = 5

3 - 2 = 1

3 \* 2 = 6

3 / 2 = 1

#include <stdio.h>

int main()

{

int A;

int B;

int X, Y, Z, W;

scanf("%d %d", &A, &B);

X = A + B;

Y = A - B;

Z = A \* B;

W = A / B;

printf("%d + %d = %d\n" ,A, B, X);

printf("%d - %d = %d\n", A, B, Y);

printf("%d \* %d = %d\n", A, B, Z);

printf("%d / %d = %d\n", A, B, W);

return 0;

}

**02-1. 厘米换算英尺英寸(15)**

如果已知英制长度的英尺foot和英寸inch的值，那么对应的米是(foot+inch/12)\*0.3048。现在，如果用户输入的是厘米数，那么对应英制长度的英尺和英寸是多少呢？别忘了1英尺等于12英寸。

****输入格式：****

输入在一行中给出1个正整数，单位是厘米。

****输出格式：****

在一行中输出这个厘米数对应英制长度的英尺和英寸的整数值，中间用空格分开。

****输入样例：****

170

****输出样例：****

5 6

#include <stdio.h>

int main()

{

int cm;

scanf("%d", &cm);

int foot = cm / 30.48;

int inch = (cm/30.48 - foot) \* 12;

printf("%d %d", foot, inch);

return 0;

}

**02-2. 然后是几点(15)**

有时候人们用四位数字表示一个时间，比如1106表示11点零6分。现在，你的程序要根据起始时间和流逝的时间计算出终止时间。读入两个数字，第一个数字以这样的四位数字表示当前时间，第二个数字表示分钟数，计算当前时间经过那么多分钟后是几点，结果也表示为四位数字。当小时为个位数时，没有前导的零，即5点30分表示为530。注意，第二个数字表示的分钟数可能超过60，也可能是负数。

****输入格式：****

输入在一行中给出2个整数，分别是四位数字表示的起始时间、以及流逝的分钟数，其间以空格分隔。注意：在起始时间中，当小时为个位数时，没有前导的零，即5点30分表示为530；流逝的分钟数可能超过60，也可能是负数。

****输出格式：****

输出四位数字表示的终止时间。题目保证起始时间和终止时间在同一天内。

****输入样例：****

1120 110

****输出样例：****

1310

#include <stdio.h>

int main()

{

int i, j, x = 0, y, z;

scanf("%d%d", &i, &j);

if(i >= 1000)

{

x = i / 1000;

y = (i % 1000) / 100;

z = (i %1000) %100;

}

else

{

y = i / 100;

z = i % 100;

}

int s = (10\*x + y) \* 60 + z;

int n = s + j;

i = n / 60 \*100 + (n - ((n / 60 )\* 60));

printf("%d", i);

return 0;

}

**02-3. 逆序的三位数(10)**

程序每次读入一个正3位数，然后输出按位逆序的数字。注意：当输入的数字含有结尾的0时，输出不应带有前导的0。比如输入700，输出应该是7。

****输入格式：****

每个测试是一个3位的正整数。

****输出格式：****

输出按位逆序的数。

****输入样例：****

123

****输出样例：****

321

#include <stdio.h>

int main()

{

int i;

scanf("%d", &i);

int a = i % 100 %10;

int b = i / 100;

int c = i %100 / 10;

i = a\*100 + b + c\*10;

printf("%d", i);

return 0;

}

# **02-4. BCD解密(10)**

BCD数是用一个字节来表达两位十进制的数，每四个比特表示一位。所以如果一个BCD数的十六进制是0x12，它表达的就是十进制的12。但是小明没学过BCD，把所有的BCD数都当作二进制数转换成十进制输出了。于是BCD的0x12被输出成了十进制的18了！

现在，你的程序要读入这个错误的十进制数，然后输出正确的十进制数。提示：你可以把18转换回0x12，然后再转换回12。

****输入格式：****

输入在一行中给出一个[0, 153]范围内的正整数，保证能转换回有效的BCD数，也就是说这个整数转换成十六进制时不会出现A-F的数字。

****输出格式：****

输出对应的十进制数。

****输入样例：****

18

****输出样例：****

12

#include <stdio.h>

int main()

{

int i;

scanf("%d", &i);

int t = i % 16 + i / 16 \* 10;

printf("%d", t);

return 0;

}

**03-0. 超速判断(10)**

模拟交通警察的雷达测速仪。输入汽车速度，如果速度超出60 mph，则显示“Speeding”，否则显示“OK”。

****输入格式：****

输入在一行中给出1个不超过500的非负整数，即雷达测到的车速。

****输出格式：****

在一行中输出测速仪显示结果，格式为：“Speed: V - S”，其中V是车速，S或者是Speeding、或者是OK。

****输入样例1：****

40

****输出样例1：****

Speed: 40 - OK

****输入样例2：****

75

****输出样例2：****

Speed: 75 - Speeding

#include <stdio.h>

int main()

{

int i;

scanf("%d", &i);

if(i <= 60)

printf("Speed: %d - OK", i);

else

printf("Speed: %d - Speeding", i);

return 0;

}

# **03-1. 三天打鱼两天晒网(15)**

中国有句俗语叫“三天打鱼两天晒网”。假设某人从某天起，开始“三天打鱼两天晒网”，问这个人在以后的第N天中是“打鱼”还是“晒网”？

****输入格式：****

输入在一行中给出1个不超过1000的正整数N。

****输出格式：****

在一行中输出此人在第N天中是“Fishing”（即“打鱼”）还是“Drying”（即“晒网”），并且输出“in day N”。

****输入样例1：****

103

****输出样例1：****

Fishing in day 103

****输入样例2：****

34

****输出样例2：****

Drying in day 34

#include <stdio.h>

int main()

{

int i;

scanf("%d", &i);

if(i%5 <= 3 && i%5 != 0)

printf("Fishing in day %d", i);

else

printf("Drying in day %d", i);

return 0;

}

# **03-2. 用天平找小球(10)**

三个球A、B、C，大小形状相同且其中有一个球与其他球重量不同。要求找出这个不一样的球。

****输入格式：****

输入在一行中给出3个正整数，顺序对应球A、B、C的重量。

****输出格式：****

在一行中输出唯一的那个不一样的球。

****输入样例：****

1 1 2

****输出样例：****

C

# include <stdio.h>

int main()

{

int A, B, C;

scanf("%d%d%d",&A, &B, &C);

if(A != B && B ==C)

printf("A");

else if(B != A && A == C)

printf("B");

else if(C != A && A == B)

printf("C");

return 0;

}

# **03-3. 12-24小时制(15)**

编写一个程序，要求用户输入24小时制的时间，然后显示12小时制的时间。

****输入格式：****

输入在一行中给出带有中间的“:”符号（半角的冒号）的24小时制的时间，如12:34表示12点34分。当小时或分钟数小于10时，均没有前导的零，如5:6表示5点零6分。

*提示：在scanf的格式字符串中加入“:”，让scanf来处理这个冒号。*

****输出格式：****

在一行中输出这个时间对应的12小时制的时间，数字部分格式与输入的相同，然后跟上空格，再跟上表示上午的字符串“AM”或表示下午的字符串“PM”。如“5:6 PM”表示下午5点零6分。注意，在英文的习惯中，中午12点被认为是下午，所以24小时制的12:00就是12小时制的12:0 PM；而0点被认为是第二天的时间，所以是0:0 AM。

****输入样例：****

21:11

****输出样例：****

9:11 PM

#include <stdio.h>

int main()

{

int i, j;

scanf("%d:%d", &i, &j);

if(i >= 0 && i < 12)

printf("%d:%d AM", i, j);

else if(i == 12)

printf("%d:%d PM", i, j);

else if(i == 24)

printf("%d:%d AM", i-24, j);

else

printf("%d:%d PM", i-12, j);

return 0;

}

**03-4. 成绩转换(15)**

本题要求编写程序将一个百分制成绩转换为五分制成绩。转换规则：

大于等于90分为A；小于90且大于等于80为B；小于80且大于等于70为C；小于70且大于等于60为D；小于60为E。

****输入格式：****

输入在一行中给出1个整数的百分制成绩。

****输出格式：****

在一行中输出对应的五分制成绩。

****输入样例：****

90

****输出样例：****

A

#include <stdio.h>

int main()

{

int i;

scanf("%d", &i);

if(i >= 90)

printf("A");

else if(i < 90 && i >= 80)

printf("B");

else if(i < 80 && i >= 70)

printf("C");

else if(i < 70 && i >= 60)

printf("D");

else if(i < 60)

printf("E");

return 0;

}

# **04-0. 求符合给定条件的整数集(15)**

给定不超过6的正整数A，考虑从A开始的连续4个数字。请输出所有由它们组成的无重复数字的3位数。

****输入格式：****

输入在一行中给出A。

****输出格式：****

输出满足条件的的3位数，要求从小到大，每行6个整数。整数间以空格分隔，但行末不能有多余空格。

****输入样例：****

2

****输出样例：****

234 235 243 245 253 254

324 325 342 345 352 354

423 425 432 435 452 453

523 524 532 534 542 543

#include <stdio.h>

int main()

{

int A, i, j, k, p = 0;

scanf("%d", &A);

for(i=A; i<A+4; ++i)

for(j=A; j<A+4; ++j)

for(k=A; k<A+4; ++k)

{

if(i != j && i != k && j != k)

{

printf("%d", i\*100+10\*j+k);

p++;

if(p%6 == 0)

printf("\n");

else

printf(" ");

}

}

return 0;

}

**04-1. 水仙花数(20)**

水仙花数是指一个N位正整数（N>=3），它的每个位上的数字的N次幂之和等于它本身。例如：153 = 13 + 53+ 33。本题要求编写程序,计算所有N位水仙花数。

****输入格式：****

输入在一行中给出一个正整数N（3<=N<=7）。

****输出格式：****

按递增顺序输出所有N位水仙花数，每个数字占一行。

****输入样例：****

3

****输出样例：****

153

370

371

407

#include <stdio.h>

int main()

{

int n = 0, m = 0, l = 0, k = 0, roll = 0;

int min = 1, t = 0, sum = 0, sum1 = 0;

scanf("%d", &n);

m=n;

while(m > 1) *//built the minimum for n digit*

{

min\*=10;

m--;

}

t=min;

while(t < 10\*min) *//all the number for n digit*

{

roll = t;

m = n;

sum = 1;

while(m > 0)

{

sum = sum\*(roll%10);

m--;

}

l = n;

while(l > 0)

{

roll = roll / 10;

k = roll % 10;

m = n;

sum1 = 1;

while(m > 0)

{

sum1 = sum1 \* k;

m--;

}

sum = sum + sum1;

l--;

}

if(t == sum)

{

printf("%d\n", t);

}

t++;

}

return 0;

}

# **04-2. 打印九九口诀表(15)**

下面是一个完整的下三角九九口诀表：

1\*1=1

1\*2=2 2\*2=4

1\*3=3 2\*3=6 3\*3=9

1\*4=4 2\*4=8 3\*4=12 4\*4=16

1\*5=5 2\*5=10 3\*5=15 4\*5=20 5\*5=25

1\*6=6 2\*6=12 3\*6=18 4\*6=24 5\*6=30 6\*6=36

1\*7=7 2\*7=14 3\*7=21 4\*7=28 5\*7=35 6\*7=42 7\*7=49

1\*8=8 2\*8=16 3\*8=24 4\*8=32 5\*8=40 6\*8=48 7\*8=56 8\*8=64

1\*9=9 2\*9=18 3\*9=27 4\*9=36 5\*9=45 6\*9=54 7\*9=63 8\*9=72 9\*9=81

本题要求对任意给定的1位正整数N，输出从1\*1到N\*N的部分口诀表。

****输入格式：****

输入在一行中给出一个正整数N（1<=N<=9）。

****输出格式：****

输出下三角N\*N部分口诀表，其中等号右边数字占4位、左对齐。

****输入样例：****

4

****输出样例：****

1\*1=1

1\*2=2 2\*2=4

1\*3=3 2\*3=6 3\*3=9

1\*4=4 2\*4=8 3\*4=12 4\*4=16

#include <stdio.h>

int main(void)

{

int n;

scanf("%d", &n);

int i, j;

for (i=1; i<=n; i++)

{

for (j=1; j<=i; j++)

printf("%d\*%d=%-4d", j, i, i\*j);

puts("");

}

return 0;

}

**04-3. 统计素数并求和(20)**

本题要求统计给定整数M和N区间内素数的个数并对它们求和。

****输入格式：****

输入在一行中给出2个正整数M和N（1<=M<=N<=500）。

****输出格式：****

在一行中顺序输出M和N区间内素数的个数以及它们的和，数字间以空格分隔。

****输入样例：****

10 31

****输出样例：****

7 143

#include <stdio.h>

#include <stdbool.h>

int nums(int x)

{

int i;

if(x ==1 ){

return 0;

}

for(i=2; i\*i<=x; i++){

if(x%i == 0){

return 0;

}

}

return 1;

}

int main()

{

int m,n;

int j;

scanf("%d %d", &m, &n);

int count = 0;

int sum = 0;

for(j=m; j<=n; j++)

{

if(nums(j) == 1)

{

count++;

sum += j;

}

}

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

return 0;

}

# **04-4. 猜数字游戏(15)**

猜数字游戏是令系统随机产生一个100以内的正整数，用户输入一个数对其进行猜测，需要你编写程序自动对其与随机产生的被猜数进行比较，并提示大了（“Too big”），还是小了（“Too small”），相等表示猜到了。如果猜到，则结束程序。程序还要求统计猜的次数，如果1次猜出该数，提示“Bingo!”；如果3次以内猜到该数，则提示“Lucky You!”；如果超过3次但是在N（>3）次以内（包括第N次）猜到该数，则提示“Good Guess!”；如果超过N次都没有猜到，则提示“Game Over”，并结束程序。如果在到达N次之前，用户输入了一个负数，也输出“Game Over”，并结束程序。

****输入格式：****

输入第一行中给出2个不超过100的正整数，分别是系统产生的随机数、以及猜测的最大次数N。随后每行给出一个用户的输入，直到出现负数为止。

****输出格式：****

在一行中输出每次猜测相应的结果，直到输出猜对的结果或“Game Over”则结束。

****输入样例：****

58 4

70

50

56

58

60

-2

****输出样例：****

Too big

Too small

Too small

Good Guess!

#include <stdio.h>

int main()

{

int n, times;

int i, t = 0;

scanf("%d %d", &n, ×);

while(1)

{

scanf("%d", &i);

++t;

if(i == n && t == 1)

{

printf("Bingo!\n");

break;

}

else if(i < 0 || t > times )

{

printf("Game Over");

break;

}

else if(i == n && t > 1 && t <= 3 && t <= times)

{

printf("Lucky You!\n");

break;

}

else if(i > n)

{

printf("Too big\n");

}

else if(i < n && i > 0)

{

printf("Too small\n");

}

else if(i == n && t > 3 && t <= times)

{

printf("Good Guess!\n");

break;

}

}

return 0;

}

**05-0. 求序列前N项和(15)**

本题要求编写程序,计算序列 2/1+3/2+5/3+8/5+... 的前N项之和。注意该序列从第2项起，每一项的分子是前一项分子与分母的和，分母是前一项的分子。

****输入格式：****

输入在一行中给出一个正整数N。

****输出格式：****

在一行中输出部分和的值，精确到小数点后2位。题目保证计算结果不超过双精度范围。

****输入样例：****

20

****输出样例：****

32.66

#include <stdio.h>

int main()

{

int n = 1, i;

double a = 2, b = 1, t, s = 0;

scanf("%d", &n);

for(i=1; i<=n; ++i)

{

t = a;

s += a/b;

a += b;

b = t;

}

printf("%.2lf", s);

return 0;

}

# **05-1. 约分最简分式(15)**

分数可以表示为“分子/分母”的形式。编写一个程序，要求用户输入一个分数，然后将其约分为最简分式。最简分式是指分子和分母不具有可以约分的成分了。如6/12可以被约分为1/2。当分子大于分母时，不需要表达为整数又分数的形式，即11/8还是11/8；而当分子分母相等时，仍然表达为1/1的分数形式。

****输入格式：****

输入在一行中给出一个分数，分子和分母中间以斜杠“/”分隔，如：12/34表示34分之12。分子和分母都是正整数（不包含0，如果不清楚正整数的定义的话）。

*提示：在scanf的格式字符串中加入“/”，让scanf来处理这个斜杠。*

****输出格式：****

在一行中输出这个分数对应的最简分式，格式与输入的相同，即采用“分子/分母”的形式表示分数。如5/6表示6分之5。

****输入样例：****

60/120

****输出样例：****

1/2

#include <stdio.h>

int main()

{

int i, j, n, k;

scanf("%d/%d", &i, &j);

loop:

if(i >= j)

n = j;

else

n = i;

for(k=2; k<=n; ++k)

{

if(j % k == 0 && i % k == 0)

{

j /= k;

i /= k;

goto loop;

}

}

printf("%d/%d", i, j);

return 0;

}

**05-2. 念数字(15)**

输入一个整数，输出每个数字对应的拼音。当整数为负数时，先输出“fu”字。十个数字对应的拼音如下：

0: ling

1: yi

2: er

3: san

4: si

5: wu

6: liu

7: qi

8: ba

9: jiu

****输入格式：****

输入在一行中给出一个整数，如：1234。

*提示：整数包括负数、零和正数。*

****输出格式：****

在一行中输出这个整数对应的拼音，每个数字的拼音之间用空格分开，行末没有最后的空格。如yi er san si。

****输入样例：****

-600

****输出样例：****

fu liu ling ling

#include <stdio.h>

#include <string.h>

char str[20];

const char \*num[] = {"ling", "yi", "er", "san", "si", "wu", "liu", "qi", "ba", "jiu"};

int main()

{

int i, k;

scanf("%s", str);

k = strlen(str);

if ( str[0] == '-') printf("fu");

else

printf("%s", num[str[0]-'0']);

for ( i=1; i<k; i++ )

{

printf(" %s", num[str[i]-'0']);

}

printf("\n");

return 0;

}

# **05-3. 求a的连续和(15)**

输入两个整数a和n，a的范围是[0,9]，n的范围是[1,8]，求数列之和S = a+aa+aaa+...+aaa...a（n个a）。如a为2、n为8时输出的是2+22+222+...+22222222的和。

****输入格式：****

输入在一行中给出两个整数，先后表示a和n。

****输出格式：****

在一行中输出要求的数列之和。

****输入样例：****

2 4

****输出样例：****

2468

#include <stdio.h>

#include <math.h>

int main()

{

int a, n, i, s = 0;

scanf("%d%d", &a, &n);

for(i=1; i<=n; ++i)

{

s += (pow(10.0,i)-1) \* a / 9;

}

printf("%d", s);

return 0;

}

# **06-0. 混合类型数据格式化输入(5)**

本题要求编写程序，顺序读入浮点数1、整数、字符、浮点数2，再按照字符、整数、浮点数1、浮点数2的顺序输出。

****输入格式：****

输入在一行中顺序给出浮点数1、整数、字符、浮点数2，其间以1个空格分隔。

****输出格式：****

在一行中按照字符、整数、浮点数1、浮点数2的顺序输出，其中浮点数保留小数点后2位。

****输入样例：****

2.12 88 c 4.7

****输出样例：****

c 88 2.12 4.70

#include <stdio.h>

int main()

{

float f1, f2;

char ch;

int i;

scanf("%f %d %c %f", &f1, &i, &ch, &f2);

printf("%c %d %.2f %.2f", ch, i, f1, f2);

return 0;

}

**06-1. 简单计算器(20)**

模拟简单运算器的工作。假设计算器只能进行加减乘除运算，运算数和结果都是整数，4种运算符的优先级相同，按从左到右的顺序计算。

****输入格式：****

输入在一行中给出一个四则运算算式，没有空格，且至少有一个操作数。遇等号”=”说明输入结束。

****输出格式：****

在一行中输出算式的运算结果，或者如果除法分母为0或有非法运算符，则输出错误信息“ERROR”。

****输入样例：****

1+2\*10-10/2=

****输出样例：****

10

#include <stdio.h>

int main()

{

char ch = '0';

int result, i, flag = 0;

scanf("%d", &result);

while( ch != '=' )

{

scanf("%c", &ch);

if(ch == '=')

break;

scanf("%d", &i);

if( ch == '+' )

result += i;

else if( ch == '-' )

result -= i;

else if( ch == '\*' )

result \*= i;

else if( ch == '/' )

{

if( i != 0 )

result /= i;

else

flag = 1;

}

else

flag = 1;

}

if(flag)

printf("ERROR\n");

else

printf("%d", result);

return 0;

}

# **06-2. 字符串字母大小写转换(10)**

输入一个以#结束的字符串，本题要求将小写字母全部转换成大写字母，把大写字母全部转换成小写字母，其它字符不变。

****输入格式：****

输入在一行中给出一个长度不超过40的、以#结束的非空字符串。

****输出格式：****

在一行中按照要求输出转换后的字符串。

****输入样例：****

Hello World! 123#

****输出样例：****

hELLO wORLD! 123

#include <stdio.h>

int main()

{

char c;

c = getchar();

while(c != '#')

{

if(c >= 'a' && c <= 'z')

{

c -= 32;

}

else if(c >= 'A' && c <= 'Z')

{

c += 32;

}

printf("%c", c);

c = getchar();

}

return 0;

}

# **06-3. 单词长度(15)**

你的程序要读入一行文本，其中以空格分隔为若干个单词，以‘.’结束。你要输出每个单词的长度。这里的单词与语言无关，可以包括各种符号，比如“it's”算一个单词，长度为4。注意，行中可能出现连续的空格；最后的‘.’不计算在内。

****输入格式：****

输入在一行中给出一行文本，以‘.’结束。

*提示：用scanf("%c",...);来读入一个字符，直到读到‘.’为止。*

****输出格式：****

在一行中输出这行文本对应的单词的长度，每个长度之间以空格隔开，行末没有最后的空格。

****输入样例：****

It's great to see you here.

****输出样例：****

4 5 2 3 3 4

#include <stdio.h>

int main()

{

char s[100];

int i = 0;

do

{

scanf("%c", &s[i]);

i++;

} while(s[i-1] != '.');

int j;

int cnt = 0;

int k = 0;

for(j=0; j<i; j++)

{

if(s[j] != ' ')

{

cnt++;

if(k != 0 && s[j] != '.')

{

printf(" ");

k = 0;

}

if(s[j] == '.' && s[j - 1] != ' ' && cnt != 1)

{

printf("%d", cnt-1);

}

}

else if(cnt != 0)

{

printf("%d", cnt);

cnt = 0;

k = 1;

}

}

return 0;

}

# **07-0. 写出这个数 (20)**

读入一个自然数n，计算其各位数字之和，用汉语拼音写出和的每一位数字。

****输入格式：****每个测试输入包含1个测试用例，即给出自然数n的值。这里保证n小于10100。

****输出格式：****在一行内输出n的各位数字之和的每一位，拼音数字间有1空格，但一行中最后一个拼音数字后没有空格。

****输入样例：****

1234567890987654321123456789

****输出样例：****

yi san wu

#include <stdio.h>

#include <string.h>

const char \*num[] = {"ling", "yi", "er", "san", "si", "wu", "liu", "qi", "ba", "jiu"};

int main()

{

char ch, number[10];

int s = 0, i;

while( (ch = getchar()) != '\n' )

{

s += ch-'0';

}

sprintf(number, "%d", s);

printf("%s", num[number[0]-'0']);

for ( i=1; i<strlen(number); i++ )

{

printf(" %s", num[number[i]-'0']);

}

printf("\n");

return 0;

}

# **07-1. 换个格式输出整数 (15)**

让我们用字母B来表示“百”、字母S表示“十”，用“12...n”来表示个位数字n（<10），换个格式来输出任一个不超过3位的正整数。例如234应该被输出为BBSSS1234，因为它有2个“百”、3个“十”、以及个位的4。

****输入格式：****每个测试输入包含1个测试用例，给出正整数n（<1000）。

****输出格式：****每个测试用例的输出占一行，用规定的格式输出n。

****输入样例1：****

234

****输出样例1：****

BBSSS1234

****输入样例2：****

23

****输出样例2：****

SS123

#include <stdio.h>

int main()

{

int i, j, B, S;

scanf("%d", &i);

if(i >= 1 && i < 1000)

{

if(i >= 1 && i < 10)

{

for(j=1; j<=i; ++j)

{

printf("%d", j);

}

}

if(i >= 10 && i < 100)

{

S = i / 10;

for(j=1; j<=S; ++j)

{

printf("S");

}

i = i % 10;

for(j=1; j<=i; ++j)

{

printf("%d", j);

}

}

if(i >= 100 && i < 1000)

{

B = i / 100;

for(j=1; j<=B; ++j)

{

printf("B");

}

S = i /10 % 10;

for(j=1; j<=S; ++j)

{

printf("S");

}

i = i % 10;

for(j=1; j<=i; ++j)

{

printf("%d", j);

}

}

}

return 0;

}

# **07-2. A+B和C (15)**

给定区间[-231, 231]内的3个整数A、B和C，请判断A+B是否大于C。

****输入格式：****

输入第1行给出正整数T(<=10)，是测试用例的个数。随后给出T组测试用例，每组占一行，顺序给出A、B和C。整数间以空格分隔。

****输出格式：****

对每组测试用例，在一行中输出“Case #X: true”如果A+B>C，否则输出“Case #X: false”，其中X是测试用例的编号（从1开始）。

****输入样例：****

4

1 2 3

2 3 4

2147483647 0 2147483646

0 -2147483648 -2147483647

****输出样例：****

Case #1: false

Case #2: true

Case #3: true

Case #4: false

#include <stdio.h>

int main()

{

int t, i;

long a, j, k;

scanf("%d", &t);

if(t >= 1 && t <= 10)

{

for(i=1; i<=t; ++i)

{

scanf("%ld%ld%ld", &a, &j, &k);

if(a + j > k)

{

printf("Case #%d: true\n", i);

}

else

{

printf("Case #%d: false\n", i);

}

}

}

return 0;

}

# **07-3. 数素数 (20)**

令Pi表示第i个素数。现任给两个正整数M <= N <= 104，请输出PM到PN的所有素数。

****输入格式：****

输入在一行中给出M和N，其间以空格分隔。

****输出格式：****

输出从PM到PN的所有素数，每10个数字占1行，其间以空格分隔，但行末不得有多余空格。

****输入样例：****

5 27

****输出样例：****

11 13 17 19 23 29 31 37 41 43

47 53 59 61 67 71 73 79 83 89

97 101 103

#include <stdio.h>

int a[10001];

int main()

{

int i, c = 0, j, p, m, n;

a[0] = 2;

for ( i=3; ;i+=2 )

{

p = 1;

for ( j=2; j\*j<=i; j++ )

{

if ( i % j == 0 )

{

p = 0;

break;

}

}

if ( p )

{

a[++c] = i;

}

if ( c == 10000 ) break;

}

c = 0;

scanf("%d %d", &m, &n);

for ( i=m; i<=n; i++ )

{

c++;

printf("%d", a[i-1]);

if ( c % 10 == 0 )

printf("\n");

else if ( i < n )

printf(" ");

}

if ( c % 5 != 0 )

printf("\n");

return 0;

}

# **08-0. 查找整数(10)**

本题要求从输入的N个整数中查找给定的X。如果找到，输出X的位置（从0开始数）；如果没有找到，输出“Not Found”。

****输入格式：****

输入在第1行中给出2个正整数N（<=20）和X，第2行给出N个整数。数字均不超过长整型，其间以空格分隔。

****输出格式：****

在一行中输出X的位置，或者“Not Found”。

****输入样例1：****

5 7

3 5 7 1 9

****输出样例1：****

2

****输入样例2：****

5 7

3 5 8 1 9

****输出样例2：****

Not Found

#include <stdio.h>

#define N 1000

int a[N];

int main()

{

int n, key, i;

scanf("%d%d", &n, &key);

for ( i=0; i<n; i++ )

{

scanf("%d", &a[i]);

}

for ( i=0; i<n; i++ )

{

if ( a[i] == key )

{

printf("%d\n", i);

break;

}

}

if ( i == n )

{

printf("Not Found\n");

}

return 0;

}

# **08-1. 求一批整数中出现最多的个位数字(20)**

给定一批整数，分析每个整数的每一位数字，求出现次数最多的个位数字。例如给定3个整数1234、2345、3456，其中出现最多次数的数字是3和4，均出现了3次。

****输入格式：****

输入在第1行中给出正整数N（<=1000），在第2行中给出N个不超过整型范围的正整数，数字间以空格分隔。

****输出格式：****

在一行中按格式“M: n1 n2 ...”输出，其中M是最大次数，n1、n2、……为出现次数最多的个位数字，按从小到大的顺序排列。数字间以空格分隔，但末尾不得有多余空格。

****输入样例：****

3

1234 2345 3456

****输出样例：****

3: 3 4

#include <stdio.h>

#include <string.h>

int a[1002];

int main()

{

int t, i, max = 0;

int count[10];

scanf("%d", &t);

for ( i=0; i<10; i++ )

{

count[i] = 0;

}

for ( i=0; i<t; i++ )

{

scanf("%d", &a[i]);

}

for ( i=0; i<t; i++ )

{

while (a[i])

{

count[a[i]%10]++;

a[i] /= 10;

}

}

for ( i=0; i<10; i++ )

{

if ( max < count[i] )

{

max = count[i];

}

}

printf("%d:", max);

for ( i=0; i<10; i++ )

{

if ( count[i] == max )

{

printf(" %d", i);

}

}

printf("\n");

return 0;

}

# **08-2. 求矩阵的局部极大值(15)**

给定M行N列的整数矩阵A，如果A的非边界元素A[i][j]大于相邻的上下左右4个元素，那么就称元素A[i][j]是矩阵的局部极大值。本题要求给定矩阵的全部局部极大值及其所在的位置。

****输入格式：****

输入在第1行中给出矩阵A的行数M和列数N（3<=M,N<=20）；最后M行，每行给出A在该行的N个元素的值。数字间以空格分隔。

****输出格式：****

每行按照“元素值 行号 列号”的格式输出一个局部极大值，其中行、列编号从1开始。要求按照行号递增输出；若同行有超过1个局部极大值，则该行按列号递增输出。若没有局部极大值，则输出“None 总行数 总列数”。

****输入样例1：****

4 5

1 1 1 1 1

1 3 9 3 1

1 5 3 5 1

1 1 1 1 1

****输出样例1：****

9 2 3

5 3 2

5 3 4

****输入样例2：****

3 5

1 1 1 1 1

9 3 9 9 1

1 5 3 5 1

****输出样例2：****

None 3 5

#include <stdio.h>

int a[21][21];

int main()

{

int m, n, i, j, p = 0;

scanf("%d %d", &m, &n);

for ( i=0; i<m; i++ )

{

for ( j=0; j<n; j++ )

{

scanf("%d", &a[i][j]);

}

}

for ( i=1; i<m-1; i++ )

{

for ( j=1; j<n-1; j++ )

{

if ( a[i][j] > a[i][j+1] && a[i][j] > a[i][j-1] &&

a[i][j] > a[i+1][j] && a[i][j] > a[i-1][j] )

{

printf("%d %d %d\n", a[i][j], i+1, j+1);

p = 1;

}

}

}

if ( !p )

{

printf("None %d %d\n", m, n);

}

return 0;

}

# **08-3. 组个最小数 (20)**

给定数字0-9各若干个。你可以以任意顺序排列这些数字，但必须全部使用。目标是使得最后得到的数尽可能小（注意0不能做首位）。例如：给定两个0，两个1，三个5，一个8，我们得到的最小的数就是10015558。

现给定数字，请编写程序输出能够组成的最小的数。

****输入格式：****

每个输入包含1个测试用例。每个测试用例在一行中给出10个非负整数，顺序表示我们拥有数字0、数字1、……数字9的个数。整数间用一个空格分隔。10个数字的总个数不超过50，且至少拥有1个非0的数字。

****输出格式：****

在一行中输出能够组成的最小的数。

****输入样例：****

2 2 0 0 0 3 0 0 1 0

****输出样例：****

10015558

#include <stdio.h>

int b[10];

int main()

{

int i, j;

for ( i=0; i<10; i++ )

{

b[i] = 0;

}

for ( i=0; i<10; i++ )

{

scanf("%d", &b[i]);

}

for ( i=1; i<10; i++ )

{

if (b[i])

{

printf("%d", i);

b[i]--;

break;

}

}

for ( i=0; i<10; i++ )

{

for ( j=0; j<b[i]; j++ )

{

printf("%d", i);

}

}

printf("\n");

return 0;

}

# **10-0. 说反话 (20)**

给定一句英语，要求你编写程序，将句中所有单词的顺序颠倒输出。

****输入格式：****测试输入包含一个测试用例，在一行内给出总长度不超过80的字符串。字符串由若干单词和若干空格组成，其中单词是由英文字母（大小写有区分）组成的字符串，单词之间用1个空格分开，输入保证句子末尾没有多余的空格。

****输出格式：****每个测试用例的输出占一行，输出倒序后的句子。

****输入样例：****

Hello World Here I Come

****输出样例：****

Come I Here World Hello

#include <stdio.h>

#include <string.h>

char str[81];

int main()

{

int i, k;

char \*p;

gets(str);

k = strlen(str);

p = str + k;

while (1)

{

if ( p == str )

{

printf("%s\n", p);

break;

}

if (\*p == ' ' && \*(p+1) != ' ')

{

\*p = '\0';

printf("%s ", p+1);

}

p--;

}

return 0;

}

# **10-1. 在字符串中查找指定字符(15)**

输入一个字符串S，再输入一个字符c，要求在字符串S中查找字符c。如果找不到则输出“Not found”；若找到则输出字符串S中从c开始的所有字符。

****输入格式：****

输入在第1行中给出一个不超过80个字符长度的、以回车结束的非空字符串；在第2行中给出一个字符。

****输出格式：****

在一行中按照题目要求输出结果。

****输入样例1：****

It is a black box

b

****输出样例1：****

black box

****输入样例2：****

It is a black box

B

****输出样例2：****

Not found

#include <string.h>

#include <stdio.h>

char str[180];

int main()

{

char ch;

int i, k, p = 0;

gets(str);

k = strlen(str);

scanf("%c", &ch);

for ( i=0; i<k; i++ )

{

if ( str[i] == ch )

{

p = 1;

}

if ( p )

{

printf("%c", str[i]);

}

}

if ( !p )

{

printf("Not found\n");

}

else

{

printf("\n");

}

return 0;

}

# **10-2. 删除字符串中的子串(20)**

输入2个字符串S1和S2，要求删除字符串S1中出现的所有子串S2，即结果字符串中不能包含S2。

****输入格式：****

输入在2行中分别给出不超过80个字符长度的、以回车结束的2个非空字符串，对应S1和S2。

****输出格式：****

在一行中输出删除字符串S1中出现的所有子串S2后的结果字符串。

****输入样例：****

Tomcat is a male ccatat

cat

****输出样例：****

Tom is a male

#include <stdio.h>

#include <string.h>

char s1[81], s2[81];

int main()

{

int i, k;

gets(s1);

gets(s2);

char \*p = strstr(s1, s2);

while(1)

{

char \*p = strstr(s1, s2);

if ( p )

{

for ( i=0; i<strlen(p) - strlen(s2); i++ )

{

p[i] = p[strlen(s2) + i];

}

p[i] = '\0';

}

else

{

puts(s1);

break;

}

}

return 0;

}

# **10-3. 字符串逆序(15)**

输入一个字符串，对该字符串进行逆序，输出逆序后的字符串。

****输入格式：****

输入在一行中给出一个不超过80个字符长度的、以回车结束的非空字符串。

****输出格式：****

在一行中输出逆序后的字符串。

****输入样例：****

Hello World!

****输出样例：****

!dlroW olleH

#include <stdio.h>

#include <string.h>

int main()

{

char str[81];

int k;

gets(str);

k = strlen(str);

k -= 1;

for ( k; k>=0; k-- )

{

printf("%c", str[k]);

}

printf("\n");

return 0;

}

**10-4. 字符串循环左移(20)**

输入一个字符串和一个非负整数N，要求将字符串循环左移N次。

****输入格式：****

输入在第1行中给出一个不超过100个字符长度的、以回车结束的非空字符串；第2行给出非负整数N。

****输出格式：****

在一行中输出循环左移N次后的字符串。

****输入样例：****

Hello World!

2

****输出样例：****

llo World!He

#include <stdio.h>

#include <string.h>

char str[103];

int main()

{

int i, k, n;

gets(str);

k = strlen(str);

scanf("%d", &n);

if ( n % k == 0 )

puts(str);

else

{

for ( i=n%k; i<k; i++ )

{

printf("%c", str[i]);

}

for ( i=0; i<n%k; i++ )

{

printf("%c", str[i]);

}

printf("\n");

}

return 0;

}

**11-0. 平面向量加法(10)**

本题要求编写程序，计算两个二维平面向量的和向量。

****输入格式：****

输入在一行中按照“x1 y1 x2 y2”的格式给出两个二维平面向量V1=(x1, y1)和V2=(x2, y2)的分量。

****输出格式：****

在一行中按照“(x, y)”的格式输出和向量，坐标输出小数点后1位（注意不能输出-0.0）。

****输入样例：****

3.5 -2.7 -13.9 8.7

****输出样例：****

(-10.4, 6.0)

#include <stdio.h>

#include <math.h>

#define EPSILON 0.05

struct Vector {

double x;

double y;

};

int main(void)

{

struct Vector v1, v2, v3;

scanf("%lf%lf%lf%lf", &v1.x, &v1.y, &v2.x, &v2.y);

v3.x = v1.x + v2.x;

v3.y = v1.y + v2.y;

if(fabs(v3.x) < EPSILON)

v3.x = fabs(v3.x);

if(fabs(v3.y) < EPSILON)

v3.y = fabs(v3.y);

printf("(%.1f, %.1f)\n", v3.x, v3.y);

return 0;

}

**11-1. 通讯录的录入与显示(10)**

通讯录中的一条记录包含下述基本信息：朋友的姓名、出生日期、性别、固定电话号码、移动电话号码。本题要求编写程序，录入N条记录，并且根据要求显示任意某条记录。

****输入格式：****

输入在第1行给出正整数N（<=10）；随后N行，每行按照格式“姓名 生日 性别 固话 手机”给出一条记录。其中“姓名”是不超过10个字符、不包含空格的非空字符串；生日按“yyyy/mm/dd”的格式给出年月日；性别用“M”表示“男”、“F”表示“女”；“固话”和“手机”均为不超过15位的连续数字，前面有可能出现“+”。

在通讯录记录输入完成后，最后一行给出正整数K，并且随后给出K个整数，表示要查询的记录编号（从0到N-1顺序编号）。数字间以空格分隔。

****输出格式：****

对每一条要查询的记录编号，在一行中按照“姓名 固话 手机 性别 生日”的格式输出该记录。若要查询的记录不存在，则输出“Not Found”。

****输入样例：****

3

Chris 1984/03/10 F +86181779452 13707010007

LaoLao 1967/11/30 F 057187951100 +8618618623333

QiaoLin 1980/01/01 M 84172333 10086

2 1 7

****输出样例：****

LaoLao 057187951100 +8618618623333 F 1967/11/30

Not Found

#include <stdio.h>

#define N 10

struct Person {

char name[11];

char birthday[11];

char sex;

char fixed[17];

char mobile[17];

};

int main(void)

{

struct Person p[N];

int num[N];

int i, n, k;

scanf("%d", &n);

for(i = 0; i < n; ++i)

scanf("%s %s %c %s %s", p[i].name, p[i].birthday,

&p[i].sex, p[i].fixed, p[i].mobile);

scanf("%d", &k);

for(i = 0; i < k; ++i)

scanf("%d", &num[i]);

for(i = 0; i < k; ++i) {

if(num[i] >= 0 && num[i] < n)

printf("%s %s %s %c %s\n", p[num[i]].name,

p[num[i]].fixed, p[num[i]].mobile, p[num[i]].sex, p[num[i]].birthday);

else

printf("Not Found\n");

}

return 0;

}