寒假蓝桥杯作业一----------扎实的基础训练

1、有一群海盗（不多于20人），在船上比拼酒量。过程如下：打开一瓶酒，所有在场的人平分喝下，有几个人倒下了。再打开一瓶酒平分，又有倒下的，再次重复...... 直到开了第4瓶酒，坐着的已经所剩无几，海盗船长也在其中。当第4瓶酒平分喝下后，大家都倒下了。

等船长醒来，发现海盗船搁浅了。他在航海日志中写到：“......昨天，我正好喝了一瓶.......奉劝大家，开船不喝酒，喝酒别开船......”

请你根据这些信息，推断开始有多少人，每一轮喝下来还剩多少人。

如果有多个可能的答案，请列出所有答案，每个答案占一行。

格式是：人数,人数,...

例如,有一种可能是：20,5,4,2,0

穷举不丢人。

答案写在这。

#include <stdio.h>

#include <stdlib.h>

int main()

{

int a,b,c,d;

float sum;

for(a=2;a<=20;a++)

{

for(b=2;b<a;b++)

{

for(c=2;c<b;c++)

{

for(d=2;d<c;d++)

{

sum=1.0/a+1.0/b+1.0/c+1.0/d;

if(sum==1)

printf("%d %d %d %d 0\n",a,b,c,d);

}

}

}

}

return 0;

}

上次学长就说以后这种题不用那么多循环好点，于是这次用递归来写，至少看着厉害多了。

#include <stdio.h>

int a[5];

int fx(int n,int cnt,double x)//n为当前人数，cnt为当前喝酒轮数，x为海盗船长喝的酒

{

int i,j;

if(cnt==5&&x==1)

{

for(i=0; i<5; i++)

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

printf("\n");

}

if(cnt==5)

return 1;

a[cnt-1]=n;

x+=1.0/n;

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

fx(n-j,cnt+1,x);

}

int main()

{

int i;

for(i=5;i<=20;i++)

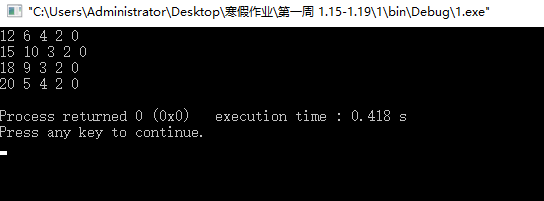
{

fx(i,1,0);

}

return 0;

}



2、猜年龄

美国数学家维纳(N.Wiener)智力早熟，11岁就上了大学。他曾在1935~1936年应邀来中国清华大学讲学。

一次，他参加某个重要会议，年轻的脸孔引人注目。于是有人询问他的年龄，他回答说：

“我年龄的立方是个4位数。我年龄的4次方是个6位数。这10个数字正好包含了从0到9这10个数字，每个都恰好出现1次。”

请你推算一下，他当时到底有多年轻。

答案写在这。

刚开始想着直接求出来年龄的立方和四次方，然后用sprintf函数将其转换为字符型，却发生了许多无法理解的错误，最后还是决定直接求出每一位数字存在数组里。。。

#include <stdio.h>

#include <math.h>

int main()

{

int age;

int age3,age4;

int i,j;

int a[10];

int flag=0;

for(age=11; age<50; age++)

{

age3=pow(age,3);

age4=pow(age,4);

if(age3>999&&age3<10000&&age4>99999&&age4<1000000)

{

i=0;

while(age3>0)

{

a[i++]=age3%10;

age3=age3/10;

}

while(age4>0)

{

a[i++]=age4%10;

age4=age4/10;

}

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

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

{

if(a[i]==a[j])

flag=1;

if(flag==1)

break;

}

if(flag==0)

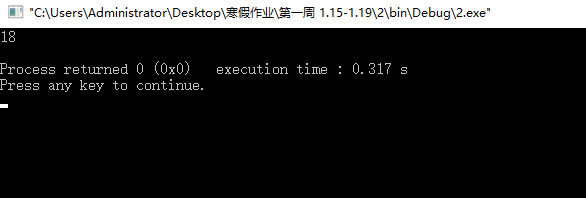
printf("%d\n",age);

}

}

return 0;

}



3、公交车票价为5角。假设每位乘客只持有两种币值的货币：5角、1元。再假设持有5角的乘客有m人，持有1元的乘客有n人。由于特殊情况，开始的时候，售票员没有零钱可找。我们想知道这m+n名乘客以什么样的顺序购票则可以顺利完成购票过程。显然，m < n的时候，无论如何都不能完成，m >=n的时候，有些情况也不行。比如，第一个购票的乘客就持有1元。下面的程序计算出这m+n名乘客所有可能顺利完成购票的不同情况的组合数目。注意：只关心5角和1元交替出现的次序的不同排列，持有同样币值的两名乘客交换位置并不算做一种新的情况来计数。

//m: 持有5角币的人数

//n: 持有1元币的人数

//返回：所有顺利完成购票过程的购票次序的种类数

int f(int m, int n)

{

if(m < n) return 0;

if(n==0) return 1;

return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

55511

5551

5515

5155

1555

1

/////////////

5551111

5155111

511551

#include <stdio.h>

#include <stdlib.h>

int f(int m, int n)

{

if(m < n) return 0;

if(n==0) return 1;

return f(m,n-1)+f(m-1,n);

}

int main()

{

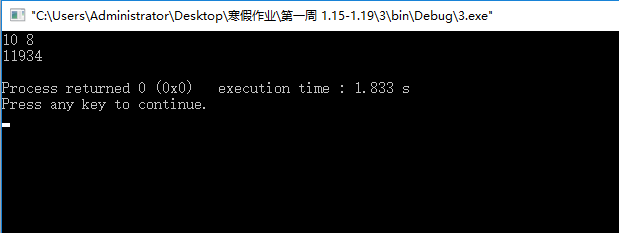
int m,n;

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

printf("%d\n",f(m,n));

return 0;

}



小贴士：

1. 浮点的非精确比较。不能用==

a. 小于一个临界值

b. 转化为整数

2. 有理数的高精度表示

注意：舍入规则的细节

有效数字位数

3.有理数的整数分数表示法

4. IEEE754 规则

特殊的有理数值

NaN

Infinite

-Infinite

0.0

-0.0

4、格子中输出

StringInGrid函数会在一个指定大小的格子中打印指定的字符串。

要求字符串在水平、垂直两个方向上都居中。

如果字符串太长，就截断。

如果不能恰好居中，可以稍稍偏左或者偏上一点。

下面的程序实现这个逻辑，请填写划线部分缺少的代码。

#include <stdio.h>

#include <string.h>

void StringInGrid(int width, int height, const char\* s)

{

int i,k;

char buf[1000];

strcpy(buf, s);

if(strlen(s)>width-2) buf[width-2]=0;

printf("+");

for(i=0;i<width-2;i++) printf("-");

printf("+\n");

for(k=1; k<(height-1)/2;k++){

printf("|");

for(i=0;i<width-2;i++) printf(" ");

printf("|\n");

}

printf("|");

printf("%\*s%s%\*s",\_(width-strlen(s)-2)/2," ",s,(width-strlen(s)-2)/2," ")\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_); //填空

printf("|\n");

for(k=(height-1)/2+1; k<height-1; k++){

printf("|");

for(i=0;i<width-2;i++) printf(" ");

printf("|\n");

}

printf("+");

for(i=0;i<width-2;i++) printf("-");

printf("+\n");

}

int main()

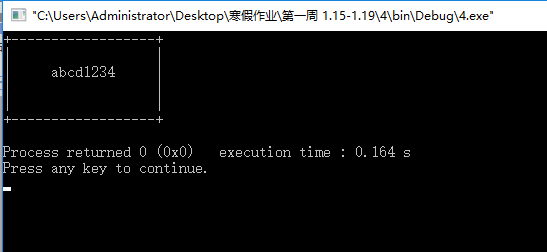
{

StringInGrid(20,6,"abcd1234");

return 0;

}

填空题，知识点在于printf中%\*s的用法，还要注意行首和行尾的两个|



对于题目中数据，应该输出：

+------------------+

| |

| abcd1234 |

| |

| |

+------------------+

（如果出现对齐问题，参看【图1.jpg】）

注意：只填写缺少的内容，不要书写任何题面已有代码或说明性文字。

5、阶乘位数

9的阶乘等于：362880

它的二进制表示为：1011000100110000000

这个数字共有19位。

请你计算，9999 的阶乘的二进制表示一共有多少位？

注意：需要提交的是一个整数，不要填写任何无关内容（比如说明解释等）

6、

古罗马帝国开创了辉煌的人类文明，但他们的数字表示法的确有些繁琐，尤其在表示大数的时候，现在看起来简直不能忍受，所以在现代很少使用了。之所以这样，不是因为发明表示法的人的智力的问题，而是因为一个宗教的原因，当时的宗教禁止在数字中出现0的概念！

罗马数字的表示主要依赖以下几个基本符号：

I 1

V 5

X 10

L 50

C 100

D 500

M 1000

这里，我们只介绍一下1000以内的数字的表示法。

单个符号重复多少次，就表示多少倍。最多重复3次。比如：CCC表示300 XX表示20，但150并不用LLL表示，这个规则仅适用于I X C M。

如果相邻级别的大单位在右，小单位在左，表示大单位中扣除小单位。比如：IX表示9 IV表示4 XL表示40 更多的示例参见下表，你找到规律了吗？

I,1

II，2

III，3

IV，4

V，5

VI，6

VII，7

VIII，8

IX，9

X，10

XI，11

XII，12

XIII,13

XIV,14

XV,15

XVI,16

XVII,17

XVIII,18

XIX,19

XX,20

XXI,21

XXII,22

XXIX,29

XXX,30

XXXIV,34

XXXV,35

XXXIX,39

XL,40

L,50

LI,51

LV,55

LX,60

LXV,65

LXXX,80

XC,90

XCIII,93

XCV,95

XCVIII,98

XCIX,99

C,100

CC,200

CCC,300

CD,400

D,500

DC,600

DCC,700

DCCC,800

CM,900

CMXCIX,999

本题目的要求是：请编写程序，由用户输入若干个罗马数字串，程序输出对应的十进制表示。

输入格式是：第一行是整数n,表示接下来有n个罗马数字(n<100)。以后每行一个罗马数字。罗马数字大小不超过999。

要求程序输出n行，就是罗马数字对应的十进制数据。

例如，用户输入：

3

LXXX

XCIII

DCCII

则程序应该输出：

80

93

702

-----------------------

遇到 I sum += 1

遇到 X sum += 10

遇到 V sum += 5

.if(包含 "IX") sum -= 2;

if(包含 "IV") sum -= 2;

if(包含 "XC") sum -= 20;

...

罗马数字，真心好复杂好烦的东西，好在这道题是给定罗马数字求阿拉伯数字，按照题目提醒做即可。若是反过来就麻烦多了，各种规定.......

/\*需要知道罗马数字的表示规律，题目已经介绍的比较清楚，

然后就是把每个符号对应整数，根据规律进行求和

\*/

#include <stdio.h>

#include <string.h>

int main()

{

char a[7]={'I','V','X','L','C','D','M'};

int b[7]={1,5,10,50,100,500,1000};

int t;

scanf("%d",&t);

getchar();

while(t--)

{

char str[100];

gets(str);

int i,j;

int c[100];

for(i=0;i<strlen(str);i++)

for(j=0;j<7;j++)

{

if(str[i]==a[j])

c[i]=b[j];

}

int sum;

sum=c[0];

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

{

if(c[i]>c[i-1])//后面的大于前面的

sum+=c[i]-2\*c[i-1];

if(c[i]<=c[i-1])//后面小于等于前面的

sum+=c[i];

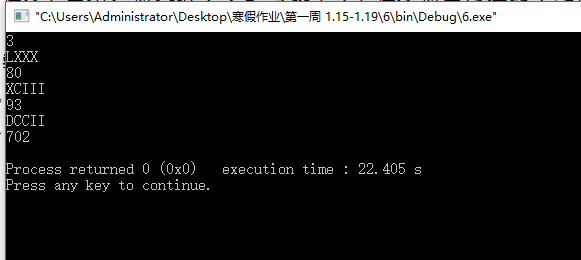
}

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

}

return 0;

}



7、A、B、C、D、E、F、G、H、I、J 共10名学生有可能参加本次计算机竞赛，也可能不参加。因为某种原因，他们是否参赛受到下列条件的约束：

1. 如果A参加，B也参加；

2. 如果C不参加，D也不参加；

3. A和C中只能有一个人参加；

4. B和D中有且仅有一个人参加；

5. D、E、F、G、H 中至少有2人参加；

6. C和G或者都参加，或者都不参加；

7. C、E、G、I中至多只能2人参加

8. 如果E参加，那么F和G也都参加。

9. 如果F参加，G、H就不能参加

10. 如果I、J都不参加，H必须参加

请编程根据这些条件判断这10名同学中参赛者名单。如果有多种可能，则输出所有的可能情况。每种情况占一行。参赛同学按字母升序排列，用空格分隔。

比如：

C D G J

就是一种可能的情况。

多种情况的前后顺序不重要

技巧： 参加为1，不参加为0，不用boolean类型

3 if(a+c<=1)

4 if(b+d==1)

离散数学中的逻辑推理题，看到题的第一反应是这些同学毛病真多。。。。。

10个判断条件，不过这道题还是挺有意思的。

比如8. 如果E参加，那么F和G也都参加。

9. 如果F参加，G、H就不能参加

很矛盾的逻辑，E参加了，就是有问题的，所以E不可能参加

/\*

A B C D E F G H I J

1 2 3 4 5 6 7 8 9 10

\*/

#include <stdio.h>

int a[11];

void judge()

{

if((a[1]+a[2]==2)||!a[1])

{

if((a[3]+a[4]==0)||a[3])

{

if(a[1]+a[3]<=1)

{

if(a[2]+a[4]==1)

{

if(a[4]+a[5]+a[6]+a[7]+a[8]>=2)

{

if((a[3]+a[7]==2)||(a[3]+a[7]==0))

{

if(a[3]+a[5]+a[7]+a[9]<=2)

{

if((a[5]+a[6]+a[7]==3)||(!a[5]))

{

if((a[6]&&a[6]+a[7]+a[8]==1)||(!a[6]))

{

if((a[9]+a[10]==0&&a[8])||(a[9]+a[10]>=1))

{

int i;

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

if(a[i])

printf("%c ",i+64

);

printf("\n");

}

}

}

}

}

}

}

}

}

}

}

void dfs(int k)

{

int i;

if(k==11)

{

judge();

return;

}

for(i=0; i<=1; i++)

{

a[k]=i;

dfs(k+1);

}

}

int main()

{

dfs(1);

return 0;

}

看到有个帖子专门讨论这道题，然后也有段我觉得很精简的代码，贴出来和有机会看到的大佬分享下。

#include <stdio.h>

void show(int \*x);

int judege(int \*x);

void curision(int \*x, int n);

int main()

{

int x[10];

curision(x,0);

return 0;

}

void show(int \*x)

{

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

if(x[i]>0) printf("%c ", i+'A');

putchar(10);

}

int judge(int \*x)

{

int t1= x[0]==0 || x[1]==1;

int t2= x[2]==1 || x[3]==0;

int t3= x[0] + x[2] <= 1;

int t4= x[1] + x[3] == 1;

int t5= x[3] + x[4] + x[5] + x[6] + x[7] >=2;

int t6= (x[2]+x[6]==2) || (x[2]+x[6]==0);

int t7= x[2]+x[4]+x[6]+x[8] <= 2;

int t8= x[4]==0 || (x[5] + x[6]==2);

int t9= x[5]==0 || (x[6] + x[7]==0);

int t10= (x[8]+x[9]>0) || x[7]==1;

return t1 && t2 && t3 && t3 && t4 && t5 && t6 && t7 && t8 && t9 && t10;

}

void curision(int \*x, int n)

{

if(n>=10)

{

if(judge(x)) show(x);

return;

}

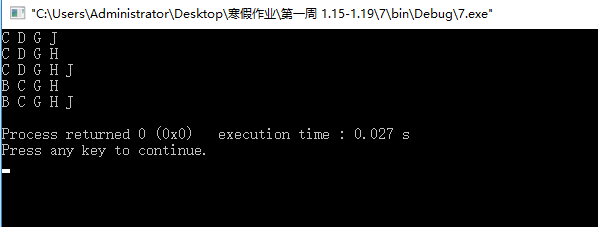
x[n]=0;

curision(x,n+1);

x[n]=1;

curision(x,n+1);

}



8、啤酒和饮料

啤酒每罐2.3元，饮料每罐1.9元。小明买了若干啤酒和饮料，一共花了82.3元。

我们还知道他买的啤酒比饮料的数量少，请你计算他买了几罐啤酒。

注意：答案是一个整数。请通过浏览器提交答案。

不要书写任何多余的内容（例如：写了饮料的数量，添加说明文字等）。

应该是做过不下三遍的题了，浮点数看着麻烦，直接夸大十倍转换为整型即可。

#include <stdio.h>

int main()

{

int a,b;

for(a=1;a<43;a++)

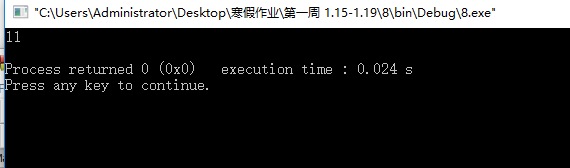
for(b=1;b<a;b++)

if(a\*19+b\*23==823)

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

return 0;

}



9、 匪警请拨110,即使手机欠费也可拨通！

为了保障社会秩序，保护人民群众生命财产安全，警察叔叔需要与罪犯斗智斗勇，因而需要经常性地进行体力训练和智力训练！

某批警察叔叔正在进行智力训练：

1 2 3 4 5 6 7 8 9 = 110;

请看上边的算式，为了使等式成立，需要在数字间填入加号或者减号（可以不填，但不能填入其它符号）。之间没有填入符号的数字组合成一个数，例如：12+34+56+7-8+9 就是一种合格的填法；123+4+5+67-89 是另一个可能的答案。

请你利用计算机的优势，帮助警察叔叔快速找到所有答案。

每个答案占一行。形如：

12+34+56+7-8+9

123+4+5+67-89

......

以前做过的题，虽然思路挺清楚，可感觉写起代码来问题还是很大，代码参靠陈大佬的，这题明天再看看。

#include<stdio.h>

int a[10];

int b[10];

int c[10];

void print(int sum)

{

int i,j;

printf("%d",c[1]);

for(i=2,j=1;i<sum+1;i++)

{

while(b[j]==1)

j++;

if(b[j]==2)

printf("+");

else

printf("-");

j++;

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

}

printf("\n");

}

void judge()

{

int i;

int sum=1,j=1,q=0;

c[1]=a[1];

for(i=1;i<9;i++)

{

if(b[i]==1)

c[sum]=c[sum]\*10+a[i+1];

else

{

c[++sum]=a[i+1];

}

}

q=c[1];

for(i=2,j=1;i<sum+1;i++)

{

while(b[j]==1)

j++;

if(b[j]==2)

q+=c[i];

else

q-=c[i];

j++;

}

if(q==110)

print(sum);

}

void dfs(int k)

{

int i;

if(k==9)

{

judge();

return;

}

for(i=1;i<4;i++)

{

b[k]=i;

dfs(k+1);

}

}

int main()

{

int i;

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

a[i]=i;

dfs(1);

return 0;

}



已知的两个答案可以输出，但不计分。

各个答案的前后顺序不重要。

注意：

请仔细调试！您的程序只有能运行出正确结果的时候才有机会得分！

请把所有类写在同一个文件中，调试好后，存入与【考生文件夹】下对应题号的“解答.txt”中即可。

相关的工程文件不要拷入。

请不要使用package语句。

源程序中只能出现JDK1.5中允许的语法或调用。不能使用1.6或更高版本。

10、消除尾一

下面的代码把一个整数的二进制表示的最右边的连续的1全部变成0

如果最后一位是0，则原数字保持不变。

如果采用代码中的测试数据，应该输出：

00000000000000000000000001100111 00000000000000000000000001100000

00000000000000000000000000001100 00000000000000000000000000001100

请仔细阅读程序，填写划线部分缺少的代码。

将最右边连续地1变成0，x与x1相与即可实现。

#include <stdio.h>

void f(int x)

{

int i;

for(i=0; i<32; i++) printf("%d", (x>>(31-i))&1);

printf(" ");

x =x&(x+1);\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

for(i=0; i<32; i++) printf("%d", (x>>(31-i))&1);

printf("\n");

}

int main()

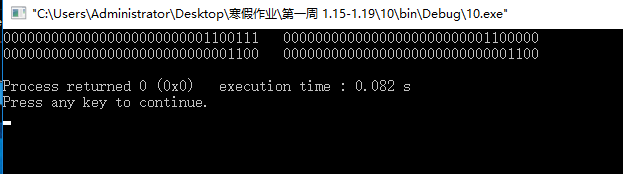
{

f(103);

f(12);

return 0;

}



注意：只填写缺少的内容，不要书写任何题面已有代码或说明性文字。

11、一步之遥

从昏迷中醒来，小明发现自己被关在X星球的废矿车里。

矿车停在平直的废弃的轨道上。

他的面前是两个按钮，分别写着“F”和“B”。

小明突然记起来，这两个按钮可以控制矿车在轨道上前进和后退。

按F，会前进97米。按B会后退127米。

透过昏暗的灯光，小明看到自己前方1米远正好有个监控探头。

他必须设法使得矿车正好停在摄像头的下方，才有机会争取同伴的援助。

或许，通过多次操作F和B可以办到。

矿车上的动力已经不太足，黄色的警示灯在默默闪烁...

每次进行 F 或 B 操作都会消耗一定的能量。

小明飞快地计算，至少要多少次操作，才能把矿车准确地停在前方1米远的地方。

请填写为了达成目标，最少需要操作的次数。

注意，需要提交的是一个整数，不要填写任何无关内容（比如：解释说明等）

以前见过的一道题目，虽然题目很长不过完全可以用一元二次方程求出解，虽然不太严谨，不过填空题，最快的求出答案最关键。

#include <stdio.h>

int main()

{

int x,y;

for(x=1; x<=200; x++)

for(y=1; y<=200; y++)

if(97\*x-127\*y==1)

{

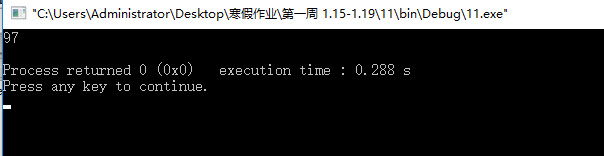
printf("%d\n",x+y);

return 0;

}

return 0;

}



12、有N个人参加100米短跑比赛。跑道为8条。程序的任务是按照尽量使每组的人数相差最少的原则分组。

例如：

N=8时，分成1组即可。

N=9时，分成2组：一组5人，一组4人。

N=25时，分4组：7、6、6、6。

请编程计算分组数字。要求从键盘输入一个正整数（1~100之间，不必考虑输入错误的情况），表示参赛的人数。程序输出每个组的人数。从大到小顺序输出，每个数字一行。

比如，用户输入：25

程序输出：

7

6

6

6

n小于8时，只有一组，人数为n

n大于8时，组数为n/8+1求出。

知道了组数也就知道了平均每组人数，然后将余数从前往后每组加一个即可

#include <stdio.h>

int main()

{

int n;

int i,x;

int a[25];

scanf("%d",&n);

if(n<=8)

{

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

return 0;

}

else

{

x=n/8+1;//需要的组数

int num=n/x;//平均每组人数

int q=n%x;//余数

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

{

if(i<q)

a[i]=num+1;

else

a[i]=num;

}

}

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

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

return 0;

}

