※循环变量是否会大一

P1035级数求和；

int k, i =1; double sum = 0.;

scanf("%d", &k);

while(sum <= k)

sum += (double) 1/i++;

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

※for()与while()在一定程度上是等价的，所以循环后是否大一与语句无关，而与代码实现有关。

For()语句中循环变量更加明显，只是在while中需要额外关注；

程序设计：

//F假设没有进入循环、直接跳出是否正确

//S出口设置的等号，根据情景与题意

※【惯用法】分解一个数d

while(d != 0)

{

c = d % 10;

d /= 10;

}

p.s.如果对循环变量进行分解，为了避免对循环控制的影响，要另设一个变量来代替i进行分解

P1424小鱼的航程问题

/\*

有一只小鱼，它上午游泳150公里，下午游泳100公里，晚上和周末都休息（实行双休日)，

假设从周x(1<=x<=7)开始算起，请问这样过了n天以后，小鱼一共累计游泳了多少公里呢？

输入格式：

输入两个整数x,n(表示从周x算起，经过n天，n在long int范围内）。

输出格式：

输出一个整数，表示小鱼累计游泳了多少公里。

\*/

#include<stdio.h>

#define DAY (150 + 100)

int main(void)

{

int x1, n, sum = 0;//基本思路：将天数截成三段：最开始的一段、中间整块的、最后末尾的段

scanf("%d %d", &x1, &n);

int z = ( n - (7 - x1 + 1) )/ 7;//中间整块的一段

int x2 = ( n - (7 - x1 + 1) ) % 7;

sum += z \* DAY \* 5;

x1 < 6 ? (sum += (7 - x1 + 1 - 2) \* DAY) : (sum += 0);//头处理

x2 < 6 ? (sum += x2 \* DAY) : (sum += 5 \* DAY); //尾处理

printf("%d", sum);

return 0;

}

/\*

另一种解法：

int main()

{

unsigned long long n,ans=0;

int x;

cin >> x >> n; //输入星期和持续天数

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

{

if((x!=6)&&(x!=7)) //只要星期不等于6和7

ans += 250; //总长度增加250

if(x==7) //当x等于7的时候

x=1; // x归位为1

else //其他情况下（x不等于7的时候）

x++; //x自加1

}

cout << ans; //输出总路程

return 0;

}

\*/

输入一串数字，然后统计每个数字出现的次数。

方案一：

int count[10] = {0};

char number[80];

int i = 0, j = 0;

printf("Enter a number: ");

while( scanf("%c", &number[i]) ==1 && number[i] != '\n')

{

while(number[i]-'0' != j++);

count[j-1]++;

j=0;//!!!!

i++;

}

方案二:

输入的数据存放在long型变量中，然后一位一位进行统计

※查找的时候index要设初值

∵你需要知道它在寻找的时候是在循环整体结束，还是满足条件后结束的；

int main(int argc, char \*argv[])

{

char \*week[WEEK+1] = {"", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"};

int i, index = -1;//很重要

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

{

if( strcmp(week[i], argv[1]) == 0)//2!! 系统1！！

{

index = i;

}

}

if(index == -1)

{

printf("Not found!\n");

}

else

{

printf("%s is %d.", week[index], index);

}

return 0;

}

P1008 三连击

/\*

将1,2, ……,9共9个数分成3组，分别组成3个三位数，且使这3个三位数构成1:2:3的比例，试求出所有满足条件的3个三位数。

\*/

int main(void)

{

int a[11] = {0};

bool flag = true;

int n;

for(int i = 192; i < 333; i++)

{

n = i;

a[n % 10] ++; a[n / 10 % 10] ++; a[n / 100] ++;

n = 2 \* i;

a[n % 10] ++; a[n / 10 % 10] ++; a[n / 100]++;

n = 3 \* i;

a[n % 10] ++; a[n / 10 % 10] ++; a[n / 100]++;

for(int j = 1; j <= 9; j++)

{

if(a[j] != 1)// a[j] == 0;

{

flag = false;

break;

}

}

if(flag)

{

printf("%d %d %d\n", i, 2\*i, 3\*i);

}

else

{

flag = true;

memset(a, 0, sizeof(a));

}

}

return 0;

}