09年

**编译器：**

#include <stdio.h>

#include <stdlib.h>

char a[100];

int index;

int A();

int B();

int C();

int A()

{

while(a[index]==' ')

index++;

if(a[index]=='x')

{

index++;

while(a[index]==' ')

index++;

return 1;

}

if(a[index]=='(')

{

index++;

while(a[index]==' ')

index++;

if(B()&&a[index]==')')

{

index++;

while(a[index]==' ')

index++;

return 1;

}

}

return 0;

}

int B()

{

return A()&&C();

}

int C()

{

if(a[index]=='+')

{

index++;

while(a[index]==' ')

index++;

if (!A())

return 0;

}

return 1;

}

int main()

{

int n;

while(scanf("%d",&n)!=EOF)

{

getchar();

while(n--)

{

index=0;

gets(a);

if(A())

printf("GOOD!\n");

else printf("BAD\n");

}

}

return 0;

}

**大写改小写：**

#include <stdio.h>

using namespace std;

#include <string.h>

int main()

{

char str[1000];

while(gets(str))

{

int len=strlen(str);

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

{

if(str[i]>='a'&&str[i]<='z')

str[i]-=32;

}

puts(str);

}

cout << "Hello world!" << endl;

return 0;

}

**分开连接：**

#include <iostream>

#include <stdio.h>

using namespace std;

char maze[101][19];

bool mark[101][19];

int n,m;

int go[4][2]={

{0,1},

{0,-1},

{1,0},

{-1,0}

};

void dfs(int x,int y)

{

for(int i=0;i<4;i++)

{

int nx=x+go[i][0];

int ny=y+go[i][1];

if(nx<1||nx>m||ny<1||ny>m)

continue;

if(maze[nx][ny]=='N')

continue;

if(mark[nx][ny]==true)

continue;

mark[nx][ny]=true;

dfs(nx,ny);

}

return ;

}

int main()

{

while(scanf("%d",&n)!=EOF)

{

while(n--)

{

scanf("%d",&m);

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

scanf("%s",maze[i]+1);

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

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

{

mark[i][j]=false;

}

int ans=0;

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

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

{

if(mark[i][j]==true) continue;

if(maze[i][j]=='N') continue;

dfs(i,j);

ans++;

}

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

}

}

return 0;

}

**和的数位：**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int n,a,b,sum;

while(scanf("%d",&n)!=EOF)

{

while(n--)

{

int num=0;

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

sum=a+b;

while(sum>=1)

{

sum=sum/10;

num++;

}

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

}

return 0;

}

return 0;

}

**计算和的数位（大数）**

#include <iostream>

#include <stdio.h>

#include <string.h>

using namespace std;

int main()

{

int n;

while(scanf("%d",&n)!=EOF)

{

while(n--)

{

char stra[100],strb[100];

int ans[100],cnt=0,c=0;

scanf("%s%s",stra,strb);

int lenA=strlen(stra),lenB=strlen(strb);

for(int i=lenA-1,j=lenB-1;i>=0&&j>=0;i--,j--)

{

ans[cnt]=stra[i]-'0'+strb[j]-'0'+c;

if(ans[cnt]>9)

{

ans[cnt]%=10;

c=1;

}

else c=0;

cnt++;

}

if(lenA>lenB)

{

int i=lenA-lenB-1;

while(i>=0)

{

ans[cnt]=stra[i]+c-'0';

if(ans[cnt]>9)

{

ans[cnt]%=10;

c=1;

}

else c=0;

cnt++;

i--;

}

if(c>0)

{

ans[cnt++]=1;

}

}

else if(lenA<lenB)

{

int i=lenB-lenA-1;

while(i>=0)

{

ans[cnt]=strb[i]+c-'0';

if(ans[cnt]>9)

{

ans[cnt]%=10;

c=1;

}

else c=0;

cnt++;

i--;

}

if(c>0)

ans[cnt++]=1;

}

else{

if(c>0)

{

ans[cnt++]=1;

}

}

//for(int i=cnt-1;i>=0;i--)

// printf("%d",ans[i]);

//printf("\n");

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

}

}

**}**

**路由器**

#include <stdio.h>

#include <stdlib.h>

struct connect

{

int a,b;

int cost;

}E[15001];

int Tree[1000];

int FindRoot(int x)

{

if(Tree[x]==-1)

return x;

else

{

int tmp=FindRoot(Tree[x]);

Tree[x]=tmp;

return tmp;

}

}

void swapAB(int \*a,int \*b)

{

int temp;

temp=\*a;

\*a=\*b;

\*b=temp;

}

int main()

{

int n,m,i,j;

while(scanf("%d",&n)!=EOF)

{

scanf("%d",&m);

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

scanf("%d%d%d",&E[i].a,&E[i].b,&E[i].cost);

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

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

{

if(E[i].cost>E[j].cost)

{

swapAB(&E[i].cost,&E[j].cost);

swapAB(&E[i].a,&E[j].a);

swapAB(&E[i].b,&E[j].b);

}

}

int ans=0;

printf("由小到大：\n");

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

{

printf("%d %d %d\n",E[i].a,E[i].b,E[i].cost);

}

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

Tree[i]=-1;

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

{

int a=FindRoot(E[i].a);

int b=FindRoot(E[i].b);

if(a!=b)

{

Tree[a]=b;

ans+=E[i].cost;

printf("第%d根 ",i);

}

}

printf("%d",ans);

}return 0;

}

**排序求第几个：**

#include <stdio.h>

#include <stdlib.h>

void swapAB(int \*a,int \*b)

{

int temp;

temp=\*a;

\*a=\*b;

\*b=temp;

}

int main()

{

int n,num,a[100];

while(scanf("%d",&n)!=EOF)

{

int i,j;

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

{

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

}

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

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

{

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

swapAB(&a[i],&a[j]);

}

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

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

break;

}

while(scanf("%d",&num)!=EOF)

{

int i,N;

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

{

scanf("%d",&N);

printf("%d\n",a[N-1]);

}

}

return 0;

}

**素数对**

#include <iostream>

#include <stdio.h>

using namespace std;

#include <math.h>

bool judge(int x)

{

if(x<=1) return false;

int bound=(int)sqrt(x)+1;

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

if(x%i==0)

return false;

return true;

}

int main()

{

int num;

while(scanf("%d",&num)!=EOF)

{

while(num--)

{

int n;

scanf("%d",&n);

int cnt=0;

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

{

if(judge(i)&&judge(n-i+1))

{

cnt++;

//printf("%d %d\n",i,n-i+1);

}

}

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

}

}

//cout << "Hello world!" << endl;

return 0;

}