1.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

#include<string.h>

char s[25],c;

void main(){

int i=0;

gets(s);

c=getchar();

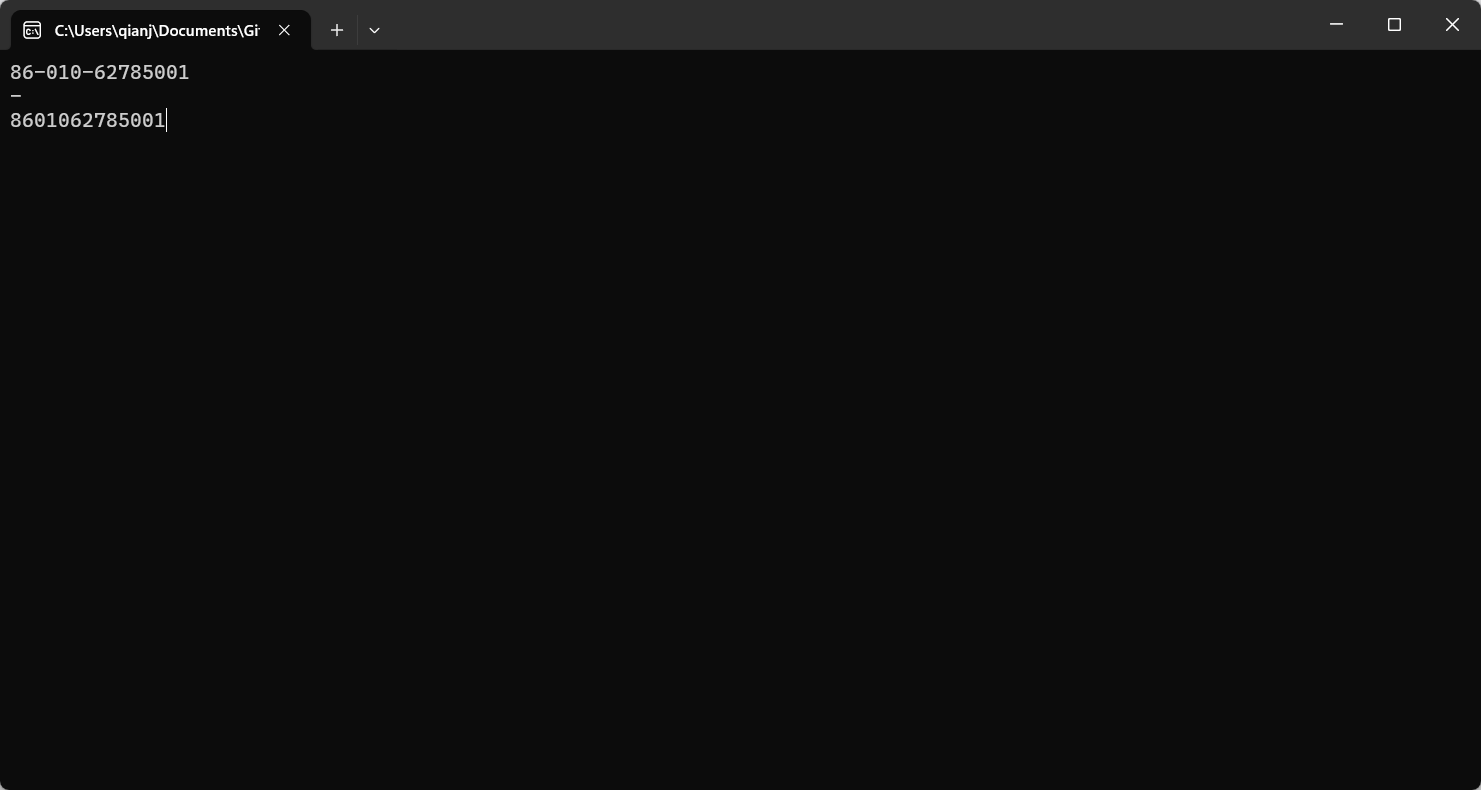
for(i=0;i<strlen(s);i++){

if(s[i]!=c) putchar(s[i]);

}

getchar();getchar();

}



2.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

#include<string.h>

char sa[25],sb[25];

void main(){

int i=0,j=0;

gets(sa);

gets(sb);

for(;i<strlen(sa)||j<strlen(sb);i++,j++){

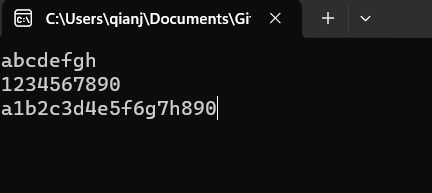
if(i<strlen(sa)) putchar(sa[i]);

if(j<strlen(sb)) putchar(sb[i]);

}

getchar();getchar();

}



3.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

int a[25],b[25],an=0,bn=0;

void main(){

int i=0,j=0;

while(scanf("%d",a+an)){

if(!a[an]) break;

++an;

}

while(scanf("%d",b+bn)){

if(!b[bn]) break;

++bn;

}

for(;i<an||j<bn;){

if(i==an){

printf("%d ",b[j]);

j++;

continue;

}

if(j==bn){

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

i++;

continue;

}

if(a[i]<b[j]){

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

i++;

}else{

printf("%d ",b[j]);

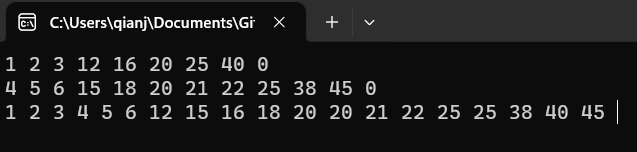
j++;

}

}

getchar();getchar();

}



4.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

#include<string.h>

#define ll long long

char s[40];

ll ans=0;

void main(){

int i=0;

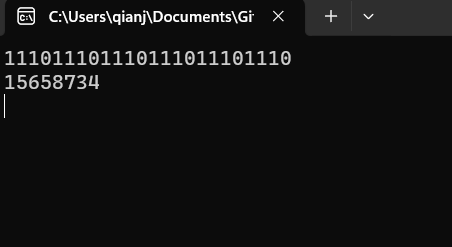
scanf("%s",s);

for(;i<strlen(s);i++) ans=(ans<<1)+s[i]-'0';

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

getchar();getchar();

}



5.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

int n;

void main(){

int i=0,j=0;

scanf("%d",&n);

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

for(j=0;i==0?j<=n:j<=i;j++){

if(!(i|j)){

if(n>=10) printf("+ ");

else printf("+ ");

}

else if(j+n>=10) printf("%-2d ",i+j);

else printf("%d ",i+j);

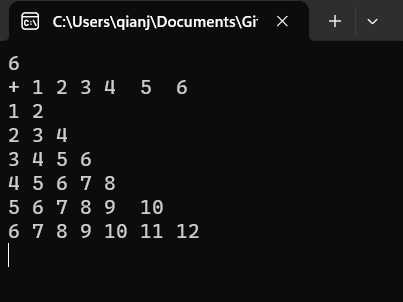
}

puts("");

}

getchar();getchar();

}



6.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

int n;

int check\_num(int a,int b);

void main(){

int i=0;

scanf("%d",&n);

for(i=0;i\*n<100000;i++){

if(check\_num(i,i\*n)){

printf("%05d/%05d=%d\n",i\*n,i,n);

}

}

getchar();getchar();

}

int check\_num(int a,int b){

int apr[10]={0};

int ok=1;

int i=0;

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

apr[a%10]=1;

a/=10;

}

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

apr[b%10]=1;

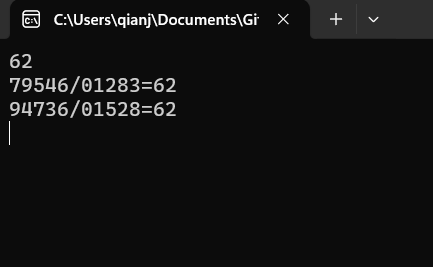
b/=10;

}

for(i=0;i<10;i++) ok&=apr[i];

return ok;

}



7.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

struct itm{

int a;

int b;

}A[405],c;

int a[25][25],n,m;

void main(){

int i,j;

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

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

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

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

A[(i-1)\*m+j-1].a=a[i][j];

}

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

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

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

}

puts("");

}

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

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

A[(i-1)\*m+j-1].b=a[i-1][j]+a[i+1][j]+a[i][j-1]+a[i][j+1];

printf("%d ",A[(i-1)\*m+j-1].b);

}

puts("");

}

for(i=0;i<n\*m;i++)

for(j=1;j<n\*m;j++){

if(A[j].b>A[j-1].b){

c=A[j];

A[j]=A[j-1];

A[j-1]=c;

}

}

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

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

printf("%d ",A[(i-1)\*m+j-1].a);

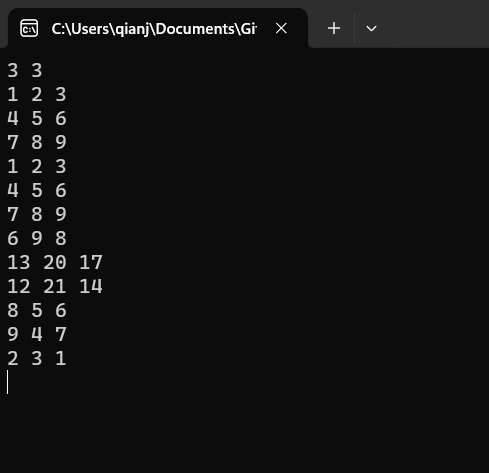
}

puts("");

}

getchar();getchar();

}



8.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

#include<string.h>

char s1[2][105],s2[15];

int n,m;

void main(){

int i,j,k;

int flag;

char\* sp;

scanf("%s%s",s1[0],s2);

m=strlen(s2);

for(k=0;;k^=1){

flag=0;

n=strlen(s1[k]);

memset(s1[k^1],0,sizeof(s1[k^1]));

sp=s1[k^1];

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

for(j=0;j<m;j++){

if(i+j>=n) break;

if(s1[k][i+j]!=s2[j]) break;

}

if(j!=m) \*sp++=s1[k][i];

else i+=m-1,flag=1;

}

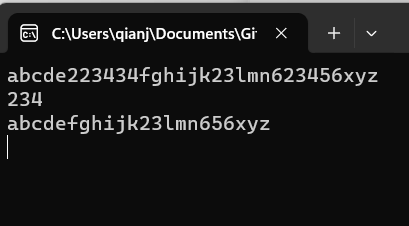
if(!flag) break;

}

puts(s1[0]);

getchar();getchar();

}



9.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

int cnt[1100];

int n;

void main(){

int i=0,tmpinput;

int has\_same=0;

int maxcnt=0,maxval=0;

scanf("%d",&n);

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

scanf("%d",&tmpinput);

cnt[tmpinput]++;

}

for(i=0;i<1025;i++){

if(cnt[i]>maxcnt) has\_same=0,maxcnt=cnt[i],maxval=i;

else if(cnt[i]==maxcnt) has\_same=1;

}

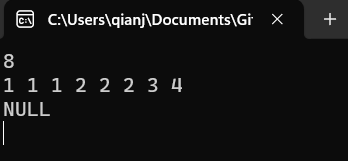
if(has\_same){

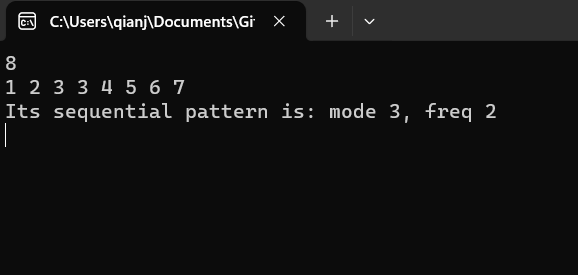
puts("NULL");

}else printf("Its sequential pattern is: mode %d, freq %d\n",maxval,maxcnt);

getchar();getchar();

}





10.

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include<stdio.h>

int n,st,m;

int is\_killed[1005];

void main(){

int loop,cnt;

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

printf("%d ",st);

is\_killed[st]=1;

++st;

for(loop=1;loop<n;loop++){

cnt=0;

for(;;st++){

if(st>n) st-=n;

cnt+=is\_killed[st]^1;

if(cnt==m) break;

}

printf("%d ",st);

is\_killed[st]=1;

}

getchar();getchar();

}

