1. Code:

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main()

{

int n=0, i=0, j, k;

char \*\*a, t[100] ;

FILE \*in, \*out ;

if ((in = fopen ("in.txt", "r")) == NULL)

{

printf("ERROR!\n") ;

exit(0) ;

}

if ((out = fopen ("out.txt", "w")) == NULL)

{

printf("ERROR!\n") ;

exit(0) ;

}

while(fgets(t, 81, in) != NULL)

n++ ;

a = (char\*\*)malloc(sizeof(char\*)\*n) ;

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

a[j] = (char\*)malloc(sizeof(char)\*81) ;

rewind(in) ;

for( ; fgets(a[i], 81, in) != NULL && i != n; i++) ;

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

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

{

if (strcmp(a[k] ,a[k+1]) > 0)

{

strcpy (t, a[k]) ;

strcpy (a[k], a[k+1]) ;

strcpy (a[k+1], t) ;

}

}

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

fputs(a[j], out) ;

fclose(in) ;

fclose(out) ;

for ( j =n-1 ; j>=0 ; j-- )

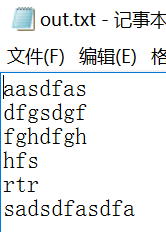
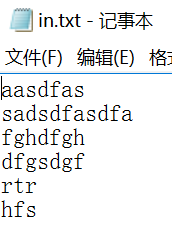
free(a[j]) ;

free(a) ;

return 0;

}

Result:



1. Code:

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

unsigned char reverse( unsigned char a )

{

unsigned char b = 0, t = 1, i, cnt = 128 ;

for ( i = 1 ; i < 8 ; i++ )

{

t = a % 2 ;

a /= 2 ;

b += cnt \* t ;

cnt /= 2 ;

}

return b ;

}

int main()

{

int n=0, j ;

unsigned char \*s ;

char scnt[1000000] ;

FILE \*fp, \*out ;

if((fp=fopen("fdata.dat","rb"))==NULL)

{

printf("Failed!");

exit(0);

}

if((out=fopen("output.dat","wb"))==NULL)

{

printf("Failed!");

exit(0);

}

while (fgets(scnt, 1000000, fp) != NULL)

n += strlen(scnt) ;

s = (unsigned char\*)malloc(sizeof(unsigned char)\*n) ;

rewind(fp) ;

fread(s, 1, n, fp) ;

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

s[j] = reverse(s[j]) ;

fwrite(s, 1, n, out) ;

fclose(fp) ;

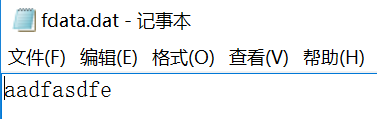
fclose(out) ;

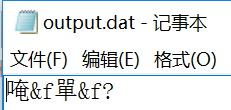
free(s) ;

return 0;

}

Result:





1. Code

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#pragma warning (disable:4996)

char c[16][3] = {"零", "一", "二", "三", "四", "五", "六", "七", "八", "九", "十", "百", "千", "万", "亿", "点"} ;

void read1( char \*s, int \*a)

{

if (a[3] != 0)

strcat(s, c[a[3]]) ;

}

void read2( char \*s, int \*a)

{

if (a[2] != 0)

{

strcat(s, c[a[2]]) ;

strcat(s, c[10]) ;

}

read1(s, a) ;

}

void read3( char \*s, int \*a)

{

if (a[1] != 0)

{

strcat(s, c[a[1]]) ;

strcat(s, c[11]) ;

}

if (a[2] != 0)

{

read2(s, a) ;

}

else if( a[3] != 0)

{

strcat(s, c[0]) ;

read1(s, a) ;

}

}

void read4(char \*s, int \*a)

{

if (a[0] != 0)

{

strcat(s, c[a[0]]) ;

strcat(s, c[12]) ;

}

if (a[1] != 0)

{

read3(s, a) ;

}

else if (a[2] != 0 || a[3] != 0)

{

strcat(s, c[0]) ;

read2(s, a) ;

}

}

void reada( char \*s, int \*a )

{

if (a[0] != 0)

read4(s, a) ;

else if (a[1] != 0)

read3(s, a) ;

else if (a[2] != 0)

read2(s, a) ;

else

read1(s, a) ;

strcat(s, "\0") ;

}

void readb(char \*s, int \*b, int \*a)

{

reada(s, b) ;

strcat(s, c[13]) ;

if (a[0] == 0 && (a[0] + a[1] + a[2] + a[3]) != 0)

strcat(s, c[0]) ;

reada(s, a) ;

}

void readc(char \*s, int \*d, int \*b, int \*a)

{

reada(s, d) ;

strcat(s, c[14]) ;

if (b[0] == 0)

{

if (!(b[1] == 0 && b[2] == 0 && b[3] == 0))

{

strcat(s, c[0]) ;

readb(s, b, a) ;

}

else if( a[0] + a[1] + a[2] + a[3] != 0)

{

strcat(s, c[0]) ;

reada(s, a) ;

}

}

else

{

readb(s, b, a) ;

}

}

void deciread( char \*s, int \*deci)

{

int i, n ;

for ( i = 14 ; i>=0 && deci[i] == 0 ; i--) ;

n = i + 1;

strcat(s, c[15]) ;

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

strcat(s, c[deci[i]]) ;

if (n == 0)

strcat(s, c[0]) ;

}

void finaltrans( char \*store, int \*flag1)

{

char s[1000], num[13], \*max = NULL, \*t, \*start, \*end, output[10000];

int \*a, \*b, \*d, n[12] = {0}, i = 0, j = 0, flag = 1, deci[15] = {0}, dflag = 0 ;

while ( store[j] < '0' || store[j] > '9')

{

if (j == strlen(store))

{

flag = 0 ;

break ;

}

j++ ;

}

if (flag == 1)

{

start = &store[j] ;

num[i] = store[j] ;

i++ ;

j++ ;

while (store[j] >= '0' && store[j] <= '9')

{

num[i] = store[j] ;

i++ ;

j++ ;

}

num[i] = '\0' ;

if (store[j] != '.')

end = &store[j] ;

else

{

dflag = 1 ;

j++ ;

i = 0 ;

while (store[j] >= '0' && store[j] <= '9')

{

deci[i] = store[j] - '0' ;

i++ ;

j++ ;

}

end = &store[j] ;

}

}

else

{

if (strstr(store,"０") != NULL)

max = strstr(store,"０") ;

if (strstr(store,"１") != NULL && ((strstr(store, "１") < max) || max == NULL) )

max = strstr(store,"１") ;

if (strstr(store,"２") != NULL && ((strstr(store, "２") < max) || max == NULL) )

max = strstr(store,"２") ;

if (strstr(store,"３") != NULL && ((strstr(store, "３") < max) || max == NULL) )

max = strstr(store,"３") ;

if (strstr(store,"４") != NULL && ((strstr(store, "４") < max) || max == NULL) )

max = strstr(store,"４") ;

if (strstr(store,"５") != NULL && ((strstr(store, "５") < max) || max == NULL) )

max = strstr(store,"５") ;

if (strstr(store,"６") != NULL && ((strstr(store, "６") < max) || max == NULL) )

max = strstr(store,"６") ;

if (strstr(store,"７") != NULL && ((strstr(store, "７") < max) || max == NULL) )

max = strstr(store,"７") ;

if (strstr(store,"８") != NULL && ((strstr(store, "８") < max) || max == NULL) )

max = strstr(store,"８") ;

if (strstr(store,"９") != NULL && ((strstr(store, "９") < max) || max == NULL) )

max = strstr(store,"９") ;

if (max == NULL)

{

\*flag1 = 0 ;

}

else

{

start = max ;

num[i] = max[1] + 80 + '0';

i++ ;

max += 2 ;

while (max[0] == -93 && max[1] >= -80 && max[1] <= -71)

{

num[i] = max[1] + 80 + '0';

i++ ;

max += 2 ;

}

num[i] = '\0' ;

if (max[0] != -93 || max[1] != -82)

end = max ;

else

{

dflag = 1 ;

max +=2 ;

i = 0 ;

while (max[0] = -93 && max[1] >= -80 && max[1] <= -71)

{

deci[i] = max[1] + 80 ;

i++ ;

max += 2 ;

}

end = max ;

}

}

}

if ( \*flag1 != 0)

{

for ( i = strlen(num) - 1, j = 11 ; i >=0 ; i--, j--)

n[j] = num[i] - '0' ;

d = &n[0] ;

b = &n[4] ;

a = &n[8];

s[0] = '\0' ;

if ((d[0] + d[1] + d[2] + d[3]) != 0)

{

if (d[2] == 1 && d[0] + d[1] == 0)

{

readc(s, d, b, a) ;

t = &s[2] ;

strcpy(s, t) ;

}

else

readc(s, d, b, a) ;

}

else if ((b[0] + b[1] + b[2] + b[3]) !=0)

{

if (b[2] == 1 && b[0] + b[1] == 0)

{

readb(s, b, a) ;

t = &s[2] ;

strcpy(s, t) ;

}

else

readb(s, b, a) ;

}

else if ((a[0] +a[1] + a[2] + a[3]) != 0)

{

if (a[2] == 1 && a[0] + a[1] == 0)

{

reada(s, a) ;

t = &s[2] ;

strcpy(s, t) ;

}

else

reada(s, a) ;

}

else

{

strcpy(s, "零") ;

}

if (dflag == 1)

deciread(s, deci) ;

strncpy(output, store, start - &store[0]) ;

output[start - &store[0]] = '\0' ;

strcat(output, s) ;

strcat(output, end) ;

strcpy(store, output) ;

}

}

int main()

{

char store[10000] ;

int flag = 1, \*flag1 ;

flag1 = &flag ;

gets(store) ;

while (flag)

finaltrans(store, flag1) ;

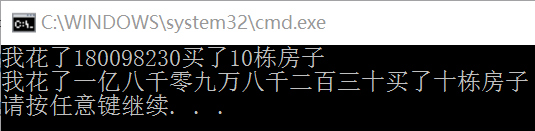
printf("%s\n", store) ;

return 0;

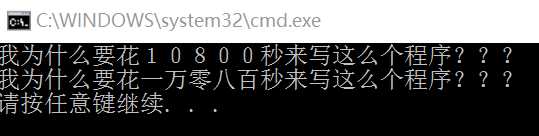
}

Result:

1.



2.



3.

