Анализ на решението на задача

СУМА

Ключовото наблюдение е, че може да получим минималната сума като заменим всички 6-ци с 5-ци в двете събираеми. По същия начин, може да получим най-голямата сума като заменим всички цифти 5 с цифрата 6 в двете дадени числа.

Има два основни начина за обработката на дадените числа:

- да се работи с масив от символи или стрингове, в които директно да се подменят петиците и шестиците, след това да се превърнат в числа и да се пресметнат двете суми. Това решение е дадено във файла suma2.cpp.

- вторият начин чете числата в променливи от тип int и прилага алгоритъма за отделяне цифрите на число, за да се подменят 5-ците с 6-ци и обратно.

Този начин е използван при основното авторово решение. Съставени са три функции obr, smallnumber, bignumber, които се извикват в главната програма по два пъти, за да се преобразуват и двете числа:

int br0;

int obr(int A)

{

int N=0,c, num;

num=A;

c=num%10;

while(c==0){

br0++;

num/=10;

c=num%10;

}

while(num>0){

c=num%10;

N=N\*10+c;

num/=10;

}

return N;

}

int smallnumber(int A)

{

int N=0,c,num;

num=A;

while(num>0){

c=num%10;

if(c==6)c=5;

N=N\*10+c;

num/=10;

}

return N;

}

int bignumber(int A)

{

int N=0,c,num;

num=A;

while(num>0){

c=num%10;

if(c==5)c=6;

N=N\*10+c;

num/=10;

}

return N;

}

int main ()

{

int A,B,minsum, maxsum,smallA,bigA,smallB,bigB,A1,B1;

cin>>A>>B;

br0=0;

A1=obr(A);

smallA= smallnumber(A1);

bigA= bignumber(A1);

for (int i=0;i<br0;i++){

smallA\*=10;

bigA\*=10;

}

br0=0;

B1=obr(B);

smallB= smallnumber(B1);

bigB= bignumber(B1);

for (int i=0;i<br0;i++){

smallB\*=10;

bigB\*=10;

}

minsum=smallA+smallB;

maxsum=bigA+bigB;

cout<<minsum<<' '<<maxsum<<endl;

}

*Автор: Пламенка Христова*