Írj olyan programot, amely egy a felhasználó által a penz.txt fájlban elhelyezett összegre megmondja, hogy hogy lehet a legkevesebb érmével kifizetni. Feltesszük, hogy 1, 2, 5, 10, 20, 50, 100 és 200 Ft-os érméink vannak. A kiírásnál azok a pénzérmék ne szerepeljenek, amikből 0 db kellene.

Példa:

// a program beolvassa a penz.txt tartalmát, vagyis egy darab egész számot

// tegyük föl, hogy a szám 1737

// a program kikalkulálja, hogy lehet a legkevesebb érmével kifizetni

8 db 200Ft-os

1 db 100Ft-os

1 db 20Ft-os

1 db 10Ft-os

1 db 5Ft-os

1 db 2Ft-os

A program a példában szereplő kékkel megjelenített sorokat nem írja ki. Figyeljünk oda arra is, hogy nincs 5-ös kerekítés, mert a példában használunk 1 és 2 Ft-os érméket is.

/\*8\_házi\*/

A címletek megadása tömbbe. Vigyázni kell, mert a cimlet[0]-ba teszi az első címletet (200)!

# include<stdio.h>

# include<fstream.h>

int main()

{int i,db,osszeg,cimlet[8]={200,100,50,20,10,5,2,1};

’penz.txt’ megnyitása olvasásra.

 //Fájl tartalmának beolvasása

 FILE \*fp1;

 fp1=fopen("penz.txt","rt");

Címletekkel való műveletek. 0-tól indul a ciklus, mert a 200 a cimlet[0]-ban van.

 fscanf(fp1,"%d",&osszeg);

 fclose(fp1);

A hányados felveszi a ’db’ deklarációját (egész). Ez tehát egy egész osztás!

 //Címletek számának meghatározása

 for(i=0;i<=7;i++)

 {db=osszeg/cimlet[i];osszeg=osszeg-db\*cimlet[i];

 if (db>0) {printf("\n%d db %dFt-os \n",db,cimlet[i]);db=0;}

 }

Létező címletek kiírása.

 printf("\n");

system("pause");

return 0;

 }//main