**★★★★☆**

**題組：Problem Set Archivewith Online Judge**

**題號：147:Dollars**

**解題者：戴維佑**

**解題日期：2025年4月17日**

**題意：**

給定一金額判斷利用以下幣值(100$, 50$, 20$, 10$, 5$, 2$, 1$, 50c, 20c, 10c, 5c; 1$ = 100c)能有多少種組合，輸入金額必為5c的倍數。

**題意範例：**

輸入:

0.20

輸出:

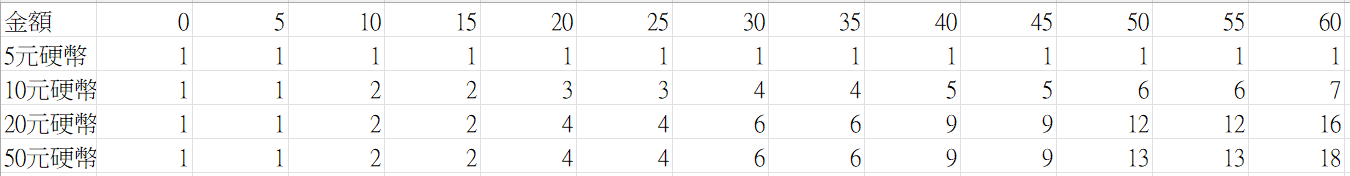
0.20 4

4種為(20c，10c\*2，10c\*1 + 5c\*2，5c\*4)

**解法：**

利用dp紀錄每個金額組合方式，當要使用幣值為n的硬幣時dp[j](金額為j時的組合方式)會等於dp[j] + dp[j-n] ，當整個dp完成後再根據要的金額從dp中提取出組合數即可。

**解法範例：**



輸入:

0.30

0.60

輸出:

0.30 6

0.60 18

**討論：**

(1)dp要用long long宣告，不然數字太大會溢位

(2)在計算dp[j]時每次可以跳5位，因為輸入一定是5c的倍數

(3)在取值時，例如3.25，要取dp[325]，要將3.25分成兩部分，前半\*100+後半，或是round(3.25\*100)，否則直接將3.25\*100會有誤差出現。

**程式：**

#include<iostream>

#include<cstdlib>

#include<iomanip>

#include<cmath>

using namespace std;

int main(){

double totalmoney;

while(cin >> totalmoney && totalmoney != 0.00){

int coin[11] = {10000, 5000, 2000, 1000, 500, 200, 100, 50, 20, 10, 5};

long long int dpcount[30005] = {1};

for(int i = 10; i >= 0; i--){

for(int j = coin[i]; j <= 30000; j+=5){

dpcount[j] = dpcount[j] + dpcount[j - coin[i]];

}

}

int tempmoney = round(totalmoney\*100);

cout << fixed << setprecision(2) << setw(6) << float(totalmoney) << setw(17) << dpcount[tempmoney] << endl;

}

return 0;

}