

/\*对于贪心算法，这里采用最优解考虑，找出用钱张数最小的一种方法，

所以就是尽可能多的使用票值最大的钱，考虑到每种钱的数量是有限的，

所以采用对当前价值钱的数量以及对钱总数除以票面对比取最小值，然

后对剩余钱进行这种处理就能达到尽可能多的使用票面大的钱，达到用

钱的数量上最少的要求,因为钱的数量是有限的,所以对最终任然存在的

money就是达不到组成这个数额进行处理，返回-1\*/

#include<iostream>

using namespace std;

#include<algorithm>

const int N=7;

int Count[N] = { 3,0,2,1,0,3,5 };

int Value[N] = { 1,2,5,10,20,50,100 };

int solve(int money)

{

int num = 0;

for (int i = N - 1; i >= 0; i--)

{

int c = min(money / Value[i], Count[i]); //调用algorithm库的min函数，函数功能返回两个数中较小的一个

money = money - c \* Value[i];

num += c;

}

if (money>0) num = -1;

return num;

}

int main()

{

int money;

cin >> money;

int res = solve(money);

if (res != -1) cout << res << endl;

else cout << "NO" << endl;

}