2.3.42 换零钱1（money）

【题目描述】

把100元换成10元、5元、1元的零钱，求每种零钱都至少各有一张的情况下，共有多少种兑换方案？

设100元可以换10元x张、5元y张、1元z张，则有：

100=10x+5y+z（$x \geqslant 1$，$y \geqslant 1$，$z \geqslant 1$）

初步分析上面的等式可以发现：$1 \leqslant x \leqslant 9$，$1 \leqslant y \leqslant 19$，$1 \leqslant z \leqslant 99$。

进一步优化可以发现：因为$x \geqslant 1$、$z \geqslant 1$，所以5y的取值范围变成$5 \leqslant 5y \leqslant 100-10\times1-1\times1$，解得$1 \leqslant y \leqslant 17$；同理，$z$的取值范围为$1 \leqslant z \leqslant 100-10\times1-5\times1$，解得$1 \leqslant z \leqslant 85$。这样使得循环体执行的次数由原来的9×19×99=16 929次缩小到9×17×85=13 005次，循环次数减少了3 924次。

```
//换零钱 — 基本算法
#include <bits/stdc++.h>
using namespace std;

int main()
{
  int Count=0;
  for (int x=1; x<=9; x++)
    for (int y=1; y<=17; y++)
      for (int z=1; z<=85; z++)
        if ((10*x+5*y+z)==100)
          Count++;
  cout<<Count<<endl;
  return 0;
}
```

还可以利用不等式减少循环次数。

在已经确定10元有x张、5元有y张的情况下，其实无须知道1元的张数，只要剩下的钱数大于0就可以组成一个方案，这样只需二重循环，即共循环9×17=153次，效率比之前提高了80多倍。

```
//换零钱 — 利用不等式优化算法
#include <bits/stdc++.h>
using namespace std;

int main()
{
  int Count=0;
  for (int x=1; x<=9; x++)
    for (int y=1; y<=17; y++)
      if (10*x+5*y<100)
        Count++;
  cout<<Count<<endl;
  return 0;
}
```

取1张10元的，则剩下的90元可以用1~17张5元和若干张1元相加得到，故共有17种方案；

取2张10元的，则剩下的80元可以用1~15张5元和若干张1元相加得到，故共有15种方案；

……

取8张10元的，则剩下的20元可以用1~3张5元和若干张1元相加得到，故共有3种方案；

取9张10元的，则剩下的10元可以用1张5元和5张1元相加得到，故共有1种方案。

综上可得，总方案数为1+3+5+…+17=81（种），则问题就变成了求9个数的和，用一重循环运算9次即可，效率比用不等式运算提高了17倍。

```
//换零钱 — 利用排列组合优化
#include <bits/stdc++.h>
using namespace std;

int main()
{
  int i,Count=0;
  for (i=1; i<=9; i++)
    Count+=2*i-1;
  cout<<Count<<endl;
  return 0;
}
```

观察数字序列1,3,5,7,9,11,…，可以发现相邻两个数的差恒为2，这显然是等差数列。记第一项为a_1，相邻两个数之间的差为公差，记为d，记第n项为a_n，容易推导出$n=(a_n-a_1)\div d+1$和$a_n=a_1+(n-1)\times d$。

求前n项和的公式如下。

$S_n=((a_1+a_n)\times n)/2$ （2-1）

$S_n=na_1+(n\times(n-1))/2\times d$ （2-2）

$S_n=na_n-(n\times(n-1))/2\times d$ （2-3）

公式（2-1）用于求知道首项、末项及项数的等差数列的前n项和。

公式（2-2）用于求知道首项、公差及项数的等差数列的前n项和。

公式（2-3）用于求知道末项、公差及项数的等差数列的前n项和。

所以程序无须循环，直接用任何一个公式就可求出和为81。

```
//换零钱 —— 利用等差数列求和公式
#include <bits/stdc++.h>
using namespace std;

int main()
{
  cout<<(1+17)*9/2<<endl;
  return 0;
}
```