

## 166 Making Change

100703004 林緯政  
 100703011 陳芸想  
 100703050 許翌君

### Topic

A interesting problem arises when goods are bought and need to be paid for, with the possibility that change may need to be given.

The problem we will be concerned with will be to minimum the number of coins that change hands at such a transaction.



### Goal

Write a program that will read in the resources available to you and the amount of the purchase and will determine the minimum number of coins that change hands.



The set of New Zealand coins comprises 5c, 10c, 20c, 50c, \$1 and \$2.

### Sample

If you have 5c\*1, 10c\*1, 20c\*2, \$1\*1.

	Pay	Change	Total
55c	$5c+10c+20c*2$	0	4
	\$1	$5c+20c*2$	4
	$5c+\$1$	50c	3



### Sample Input

- 2 4 2 2 1 0 0.95
- 2 4 2 0 1 0 0.55
- 0 0 0 0 0 0

### Sample Output

- 2
- 3

### Our strategy

1.先建一個Dynamic Programming table，記錄每種金額對最少的錢幣數量，也就是店家用的change\_table。  
 Note：店家的table 錢幣數量沒有限制。

Change Table	\$	0	5	10	15	20	25	30	35	40	45	...
	N	0	1	1	2	1	2	2	3	2	2	...

### Our strategy

2. 依據每筆測資建顧客的Dynamic Programming table，也就是顧客用的cash\_table，依照有限的錢幣數量，建立每種金額得最少錢幣數。

5c\*1, 10c\*1, 20c\*2, \$1\*1

55c

Cash Table

\$	5	10	15	20	...	45	50	55	60	...
N	1	1	2	1	...	3	3	4	0	...
\$	145	150	155							
N	4	4	5							

### Our strategy

3. 從要付金額到顧客所能支付最大金額跑一次找  
 $mincoin = cash\_table[Pay] + change\_table[Pay - Cost]$ 。

$cash\_table[55] + change\_table[55-55] = 4 + 0 = 4$   
 $cash\_table[100] + change\_table[100-55] = 1 + 3 = 4$   
 $cash\_table[105] + change\_table[105-55] = 2 + 1 = 3$

### Change Table

```
int price[6]={5,10,20,50,100,200},
change[0]=0;
for(i=5;i<=2000;i+=5){
    min=999999;
    for(j=0;j<6;j++){
        for(l=1;l+=){
            if(j-price[j]*l<0) break;
            if(change[i-price[j]*l]+l<min)
                min = change[i-price[j]*l]+l;
        }
    }
    change[i]=min;
}
```

$T(n) = O(n)$

### Cash Table

```
cash[0]=0;
for(i=0;i<6;i++){
    for(j=sum;j>=0;j--){
        for(l=1;l<=coin[i];l++){
            if(j-price[i]*l<0)
                break;
            if(cash[j]>cash[j-price[i]*l]+1)
                cash[j]=cash[j-price[i]*l]+1;
        }
    }
}
```

$T(n) = O(n)$

### Mincoin

```
for(i=number;i<=sum;i++){
    if(min<=cash[i])
        continue;
    if(min> cash[i]+change[i-number])
        min=cash[i]+change[i-number];
}
```

$T(n) = O(n)$

### TOTAL

$T(n) = O(n)$

The END