

练习2 – 零钱兑换

■ 假设有 25 分、10 分、5 分、1 分的硬币，现要找给客户 41 分的零钱，如何办到硬币个数最少？

■ 贪心策略：每一次都优先选择面值最大的硬币

① 选择 25 分的硬币，剩 16 分

② 选择 10 分的硬币，剩 6 分

③ 选择 5 分的硬币，剩 1 分

④ 选择 1 分的硬币

□ 最终的解是共 4 枚硬币

✓ 25 分、10 分、5 分、1 分硬币各一枚

```
Integer[] faces = {25, 10, 5, 1};  
Arrays.sort(faces);  
int coins = 0, money = 41;  
int idx = faces.length - 1;  
while (idx >= 0) {  
    while (money >= faces[idx]) {  
        money -= faces[idx];  
        coins++;  
    }  
    idx--;  
}
```

零钱兑换的另一个例子

■ 假设有 25 分、**20** 分、5 分、1 分的硬币，现要找给客户 41 分的零钱，如何办到硬币个数最少？

■ 贪心策略：每一步都优先选择面值最大的硬币

① 选择 25 分的硬币，剩 16 分

② 选择 5 分的硬币，剩 11 分

③ 选择 5 分的硬币，剩 6 分

④ 选择 5 分的硬币，剩 1 分

⑤ 选择 1 分的硬币

□ 最终的解是 1 枚 25 分、3 枚 5 分、1 枚 1 分的硬币，共 5 枚硬币

■ 实际上本题的最优解是：2 枚 20 分、1 枚 1 分的硬币，共 3 枚硬币