

任务进行时





1. 探索新知

1.1

数据导入



试着自己建立name、task的列表，按顺序导入特工名单和特工任务清单

```
# 数据导入
```

```
name = ['黄昏', '黎明', '007', '萧炎', '唐三']
```

```
task = ['潜入敌国获取情报',  
        '采摘生长在天山的雪莲花',  
        '给隔壁出差李阿姨家的喵喵加粮',  
        '营救被扣押的人质',  
        '破解电脑系统获取目标数据']
```

1.2

初次使用技能

1. 首先，我们需要输入特工007要使用的技能



回忆：

学过的哪个函数会提示用户输入，并等待用户输入？

【skill】：

技能；技巧；本领

```
skill = input("请使用特工技能(1.跳跃；2.隐匿；3.射击)：")
```

输入提示信息

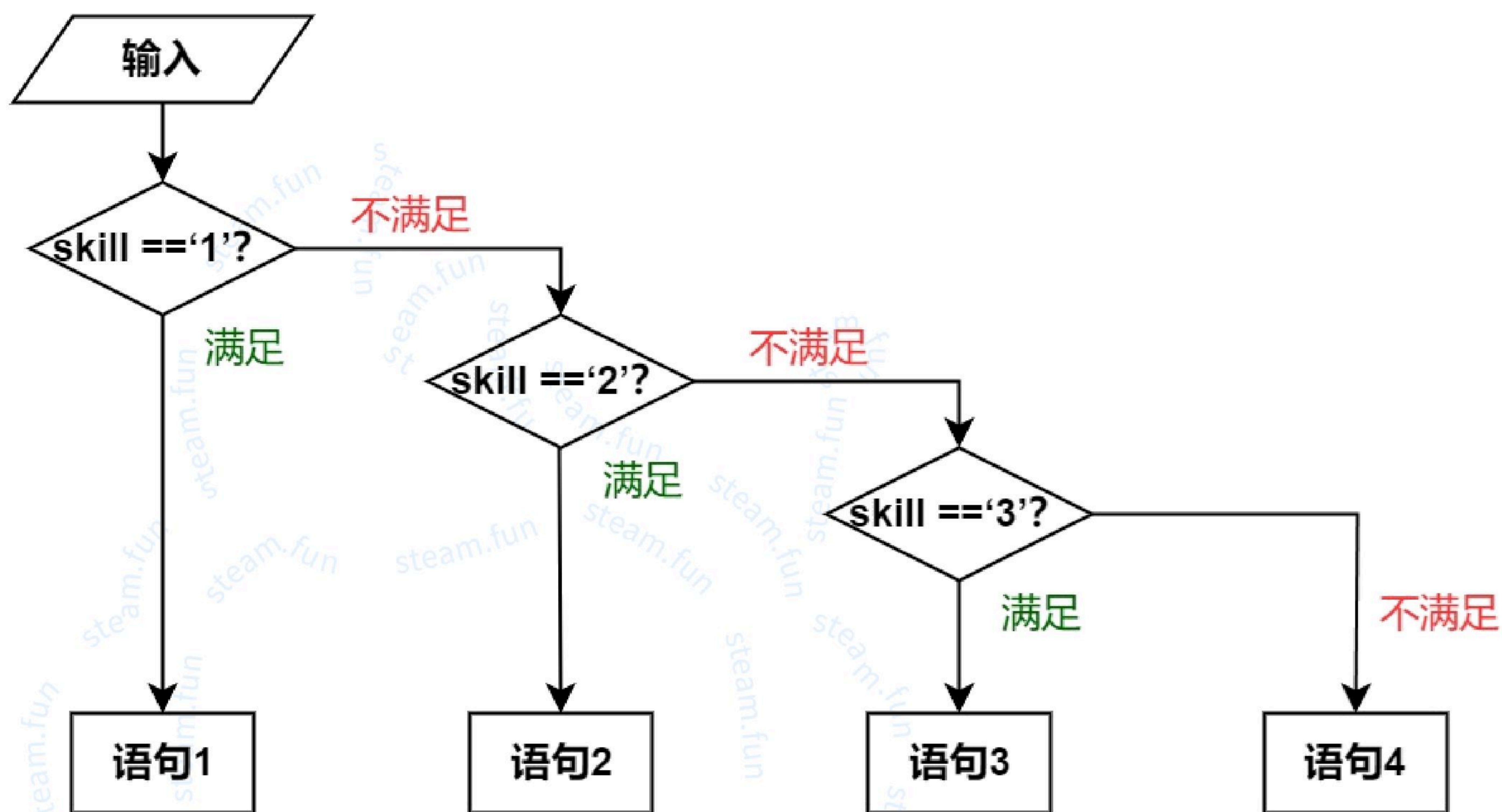
【input()函数】：

等待用户输入，将输入内容
作为字符串返回

2. 判断skill的值，执行相应操作

1.2

初次使用技能



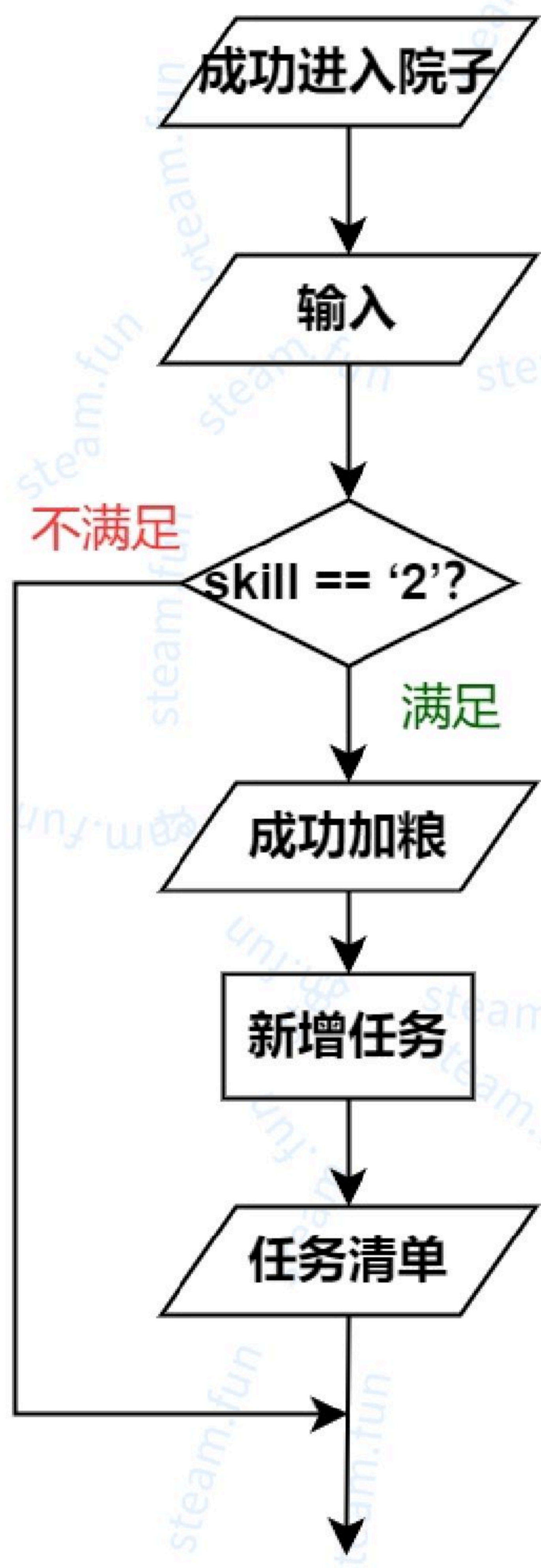
思考：

1. 该流程图对应的是什么结构？
2. 说出该结构对应的语句？
3. 为什么判断 `skill == '1'` 而不是 `skill == 1` ？

```
if skill == '1':  
    <成功进入李阿姨的院子>  
elif skill == '2':  
    print('无法进入院子')  
elif skill == '3':  
    <任务失败，被组织除名>  
else:  
    print('请正确输入技能编号')
```

• <成功进入李阿姨的院子>

成功进入院子后，任务还没有结束，继续选择合适的技能，完成给猫猫加粮的任务吧！



```
print('成功进入李阿姨的院子')
```

```
skill = input('请使用特工技能(1.跳跃;  
2.隐匿; 3.射击):')
```

```
if skill == '2':
```

```
print('成功加粮')
```



```
print(task)
```

• <新增任务>

实现方法:

1. 删除已完成任务，增加新任务（对应位置）
2. 替换已完成任务为新任务

【方法一】

【delete】：
删除

● 删除元素

- del 语句：删除指定索引处的元素

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
del fruits[2]
print(fruits)
# 输出 ['苹果', '香蕉', '葡萄']
```

【remove】：
移除

- remove(元素)方法：删除列表中指定的元素

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
fruits.remove('橙子')
print(fruits)
# 输出 ['苹果', '香蕉', '葡萄']
```

【pop】：
弹出

- pop(索引)方法：删除并返回指定索引处的元素，默认删除最后一个元素

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
fruits.pop(2)
print(fruits)
# 输出 ['苹果', '香蕉', '葡萄']
```

以上三种删除元素的方式，任意选择一种即可

007的任务密钥是2，

对应的是索引为2的任务，

内容是'给隔壁出差李阿姨家的喵喵加粮'

方法一：

```
del task[2]
```

方法二：

```
task.remove('给隔壁出差李阿姨家的喵喵加粮')
```

方法三：

```
task.pop(2)
```



接下来我们试着
在指定位置添加元素

【append】：
添加，追加

- 添加元素

➤ append(元素)方法：将元素添加到列表的末尾

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
```

```
fruits.append("樱桃")
```

```
print(fruits)
```

```
# 输出 ['苹果', '香蕉', '橙子', '葡萄', '樱桃']
```

【insert】：
插入，嵌入

➤ insert(索引, 元素)方法：在指定的索引位置插入元素

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
fruits.insert( 2, "樱桃")
print(fruits)
# 输出 ['苹果', '香蕉', '樱桃', '橙子', '葡萄']
```

【extend】：
延伸，扩大范围

➤ extend([元素1, 元素2, ...])方法：将多个元素追加到列表的末尾

```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]
fruits.extend( ["草莓", "樱桃"] )
print(fruits)
# 输出 ['苹果', '香蕉', '橙子', '葡萄', '草莓', '樱桃']
```

以上三种添加元素的方式，只有insert()方法符合需求

【方法一完整代码】

```
task.pop(2)
task.insert(2, '抢回被夺走的特工名单')
print(task)
```



接着我们思考，【方法二】
“替换已完成任务为新任务”该怎么实现呢？

【方法二】

• 修改列表元素

- 通过索引赋值：可以通过索引将一个新值赋给列表中的特定位置

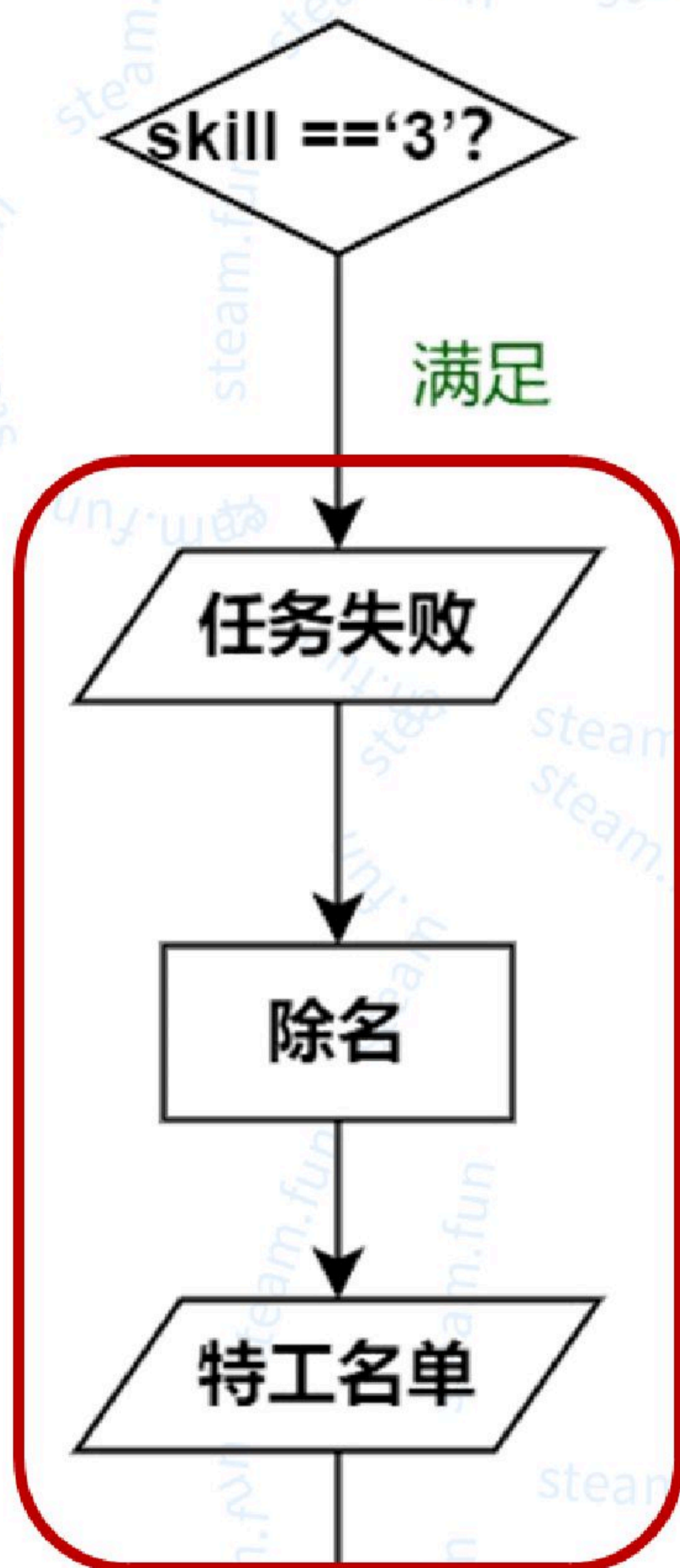
```
fruits = ["苹果", "香蕉", "橙子", "葡萄"]  
fruits[2] = '樱桃'  
print(fruits)  
# 输出 ['苹果', '香蕉', '樱桃', '葡萄']
```

【方法二完整代码】

```
task[2] = '抢回被夺走的特工名单'  
print(task)
```



自己试着根据流程图写出相应的代码吧!



提示:

1. 输出"任务失败, 被组织除名"
2. 从name列表中删除'007'
3. 输出除名后的特工名单

完整代码

```
name = ['黄昏', '黎明', '007', '萧炎', '唐三']
task = ['潜入敌国获取情报',
        '采摘生长在天山的雪莲花',
        '给隔壁出差李阿姨家的喵喵加粮',
        '营救被扣押的人质',
        '破解电脑系统获取目标数据']

skill = input("请使用特工技能(1. 跳跃; 2. 隐匿; 3. 射击): ")
if skill == '1':
    print('成功进入李阿姨的院子')
    skill = input('请使用特工技能(1. 跳跃; 2. 隐匿; 3. 射击): ')
    if skill == '2':
        print('成功加粮')
        # 方法一: 删除指定元素, 指定索引位置添加元素
        task.pop(2)
        task.insert(2, '抢回被夺走的特工名单')
        # 方法二: 修改列表元素
        # task[2] = '抢回被夺走的特工名单'
        print(task)
    elif skill == '3':
        print('无法进入院子')
    elif skill == '3':
        print('任务失败, 被组织除名')
        name.pop(2) # 或 name.remove('007'), 或 del name[2]
        print(name)
else:
    print('请正确输入技能编号')
```



2. 强化练习

1. 若要输出的列表为[2, 3, 4, 5, 5], 则横线上应填写的数字分别是? ()

A. 0, 5

B. 5, 0

C. 1, 5

D. 5, 4

```
list = [1, 2, 3, 4, 5]
```

```
list.append(____)
```

```
list.pop(____)
```

```
print(list)
```

2. 运行以下程序, 输出的结果是? ()

A. 4

B. 5

C. 8

D. 12

```
numbers=[1, 2, 3, 4]
```

```
numbers.append( [5, 6, 7, 8] )
```

```
print(len(numbers))
```

💡提示:

列表的元素可以是列表

3. 以下代码的输出结果是? ()

A. [4,2,9,1,2,3]

B. [4,3,2,9,1]

C. [4,2,9,2,1]

D. [4,2,9,3,1]

```
s=[4, 2, 9, 1]
```

```
s.insert(3, 3)
```

```
print(s)
```

4. 已知ls=[20,30,50,10,90], 以下选项中, 不能删除元素90的是? ()

A. ls.pop(-1)

B. del ls[-1]

C. ls.remove(ls[-1])

D. ls.remove(-1)



2. 强化练习

5. 下列程序执行后，打印的结果是？（ ）

A. [3,9,5]

B. [7,3,9,5]

C. [3,9,5,7]

D. [7,9,5]

```
a = [3, 9, 5]
```

```
b = a
```

```
b[0] = 7
```

```
print(b)
```



3. 术语箱

delete 删除

remove 移除

pop 弹出

append 添加, 追加

insert 插入, 嵌入

extend 延伸, 扩大范围

skill 技能, 技巧, 本领

4. 课后挑战

兔子长官

兔子长官是动物队列的负责人。她能够添加新的动物到队列中, 也能够移除不想留在队列中的动物

编写一段代码, 要求如下:

已知 `animals = ["猴子", "小花猪", "蝴蝶", "豹子"]`,

- (1) "豹子"觉得跟大家格格不入, 不想留在队列中, 请帮忙移除"豹子"
- (2) 新成员"大象"是"猴子"的好朋友, 把"大象"排到"猴子"后面 (挨着)
- (3) 队列中又来了一个新伙伴"绵羊", 让它排在队列的最后面吧
- (4) 输出新的动物队列

