

## Pattern matching

### Bài tập 1

Trên Scala, phép toán `::` dùng để nối 1 phần tử vào 1 danh sách (`3::List(2,1,5)` sẽ tạo thành danh sách `List(3,2,1,5)`). Giả sử biến `sl` đang chứa giá trị `List(3,5,6,2)`, hãy cho biết phép match sau có thành công không và nếu có thì giá trị của `h` và `t` là bao nhiêu?

`sl match {`

`case h :: tail => ....`

- ☐ Thành công, `h` là 3 và `t` là `List(5,6,2)`
- ☐ Không thành công, phải ghi case `List(3,5,6,2)` thì mới thành công
- ☐ Không thành công, phải ghi case `List(t,h)` mới thành công
- ☐ Nếu `h` đang là 3 và `t` đang là `List(5,6,2)` thì mới thành công

Chính xác

Không đúng, mặc dù ghi `List(3,5,6,2)` cũng thành công

Không đúng, nếu ghi `List(t,h)` sẽ không trùng vì `sl` là danh sách có 4 phần tử trong khi `List(t,h)` chỉ có 2 phần tử

Chưa đúng, `h` và `t` là biến tự do (chưa nhận giá trị nào) thì có thể trùng với bất kỳ giá trị nào

#### Solution

1. Correct Option
2. Wrong
3. Wrong
4. Wrong

### Bài tập 2

Trên Scala, kiểu tuple là kiểu kết hợp nhiều giá trị với nhau, ví dụ `val m = (3,4.3, True)` là một giá trị kiểu tuple kết hợp 3 kiểu `Int`, `Double` và `Boolean`. Cho biết với `m` định nghĩa trong ví dụ trên, trong phép match sau thì thành công ở case nào?

`m match {`

`case (3,1.0,True) => ...`

```
case (3,4.3,False) => ...  
  
case (_,_,_) =>  
  
}
```

- ☐ Không thành công ở case nào cả
- ☐ Thành công ở case đầu tiên vì có giá trị 3 trùng nhau
- ☐ Thành công ở case thứ hai
- ☐ Thành công ở case cuối

Chưa chính xác

Chưa chính xác, phải cả 3 giá trị trùng nhau thì mới thành công

Chưa chính xác

Chính xác, vì \_ có thể trùng với bất kỳ giá trị nào

#### Solution

1. Wrong
  2. Wrong
  3. Wrong
  4. Correct Option
-