

Written Test 1

Review Q&A

Aliasing

Call Stack vs. Catch-or-Specify Req.

NullPointerException

ArrayIndexOutOfBoundsException

Practice Written Test 1

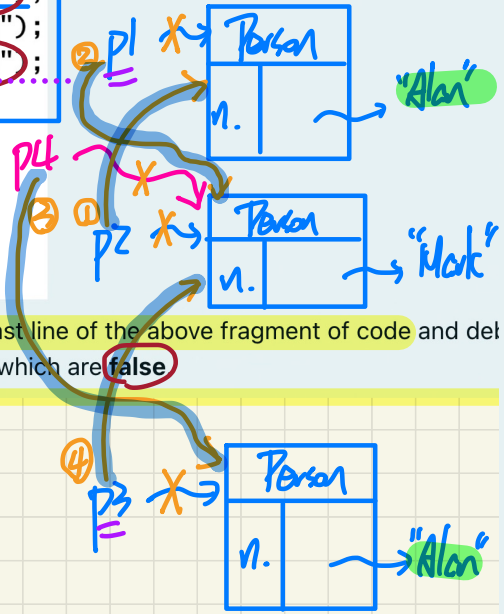
* $p1 == p3$ False
 $p1.name.equals(p3.name)$ True

Assume a `Person` class declared with: a string attribute `name` and a constructor initializing that string attribute using the input parameter.

Now consider the following fragment code which implements the `main` method of some console application class:

```
Person p1 = new Person("Alan");  
Person p2 = new Person("Mark");  
Person p3 = new Person("Alan");  
Person p4 = p2;
```

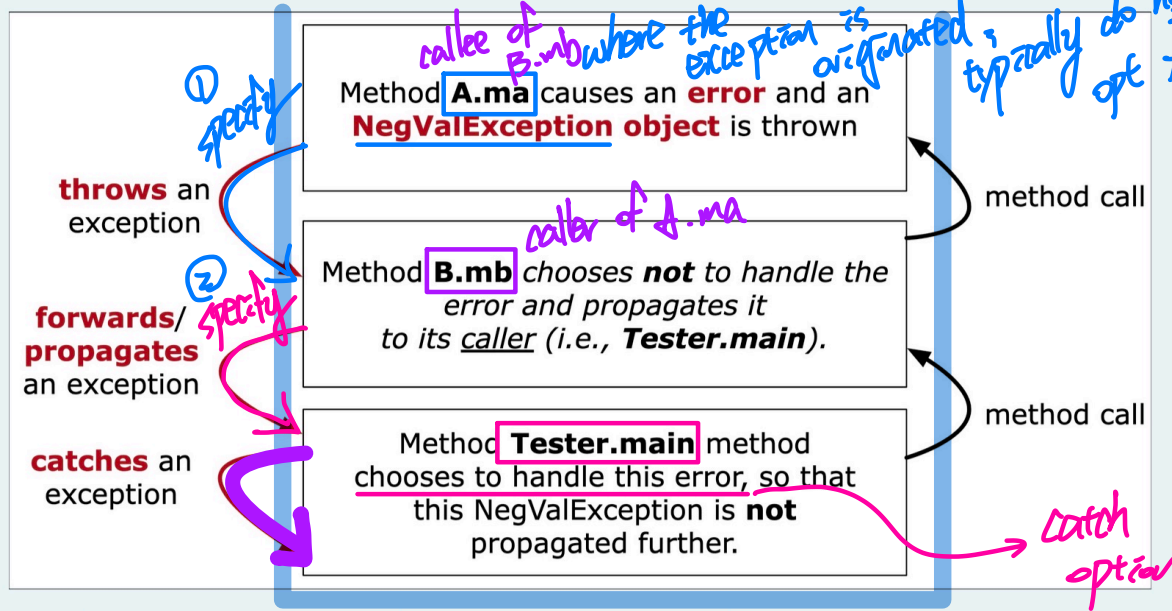
- ① $p2 = p1;$
 - ② $p1 = p4;$
 - ③ $p4 = p3;$
 - ④ $p3 = p1;$
- `System.out.println("Done!");`



Now say we place a breakpoint at the last line of the above fragment of code and del following list of statements, choose all which are false

- ☐ a. Addresses stored in p1 and p2 are the same.
- ☐ b. Addresses stored in p1 and p3 are the same.
- ☐ c. Addresses stored in p1 and p4 are the same.
- ☐ d. Addresses stored in p2 and p3 are the same.
- ☐ e. Addresses stored in p2 and p4 are the same.
- ☐ f. Addresses stored in p3 and p4 are the same.
- ☐ g. The `name` attribute value of p1 is the same as that of p2.
- ☐ h. The `name` attribute value of p1 is the same as that of p3.
- ☐ i. The `name` attribute value of p1 is the same as that of p4.
- ☐ j. The `name` attribute value of p2 is the same as that of p3.
- ☐ k. The `name` attribute value of p2 is the same as that of p4.
- ☐ l. The `name` attribute value of p3 is the same as that of p4.

Consider the following call stack where method `ma` from class `A` throws a `NegValException`:



In the above call stack, upon satisfying the catch-or-specify requirement, how many methods opt for the specify option? Your answer must be an integer value.

②.

Atypical
`ma() {`

`try {`
 `(...)`
 `throw new ...`
`} catch (...) {`
 `...`
`} else {`
 `...`
`}`

`}`

Q7

m: the first method from top of the stack that opts for the catch option

thrown by method x

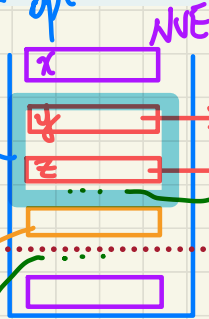
At a runtime call stack, if a method implements a try-catch block to handle a `NegValException` that may be thrown from its callee, then this method's caller is still obliged to either catch or specify that `NegValException`.

Select one:

☐ True

☐ False

all methods (callers) opt for the specify option



NVE thrown

caller of x

caller of z

calls between z and m

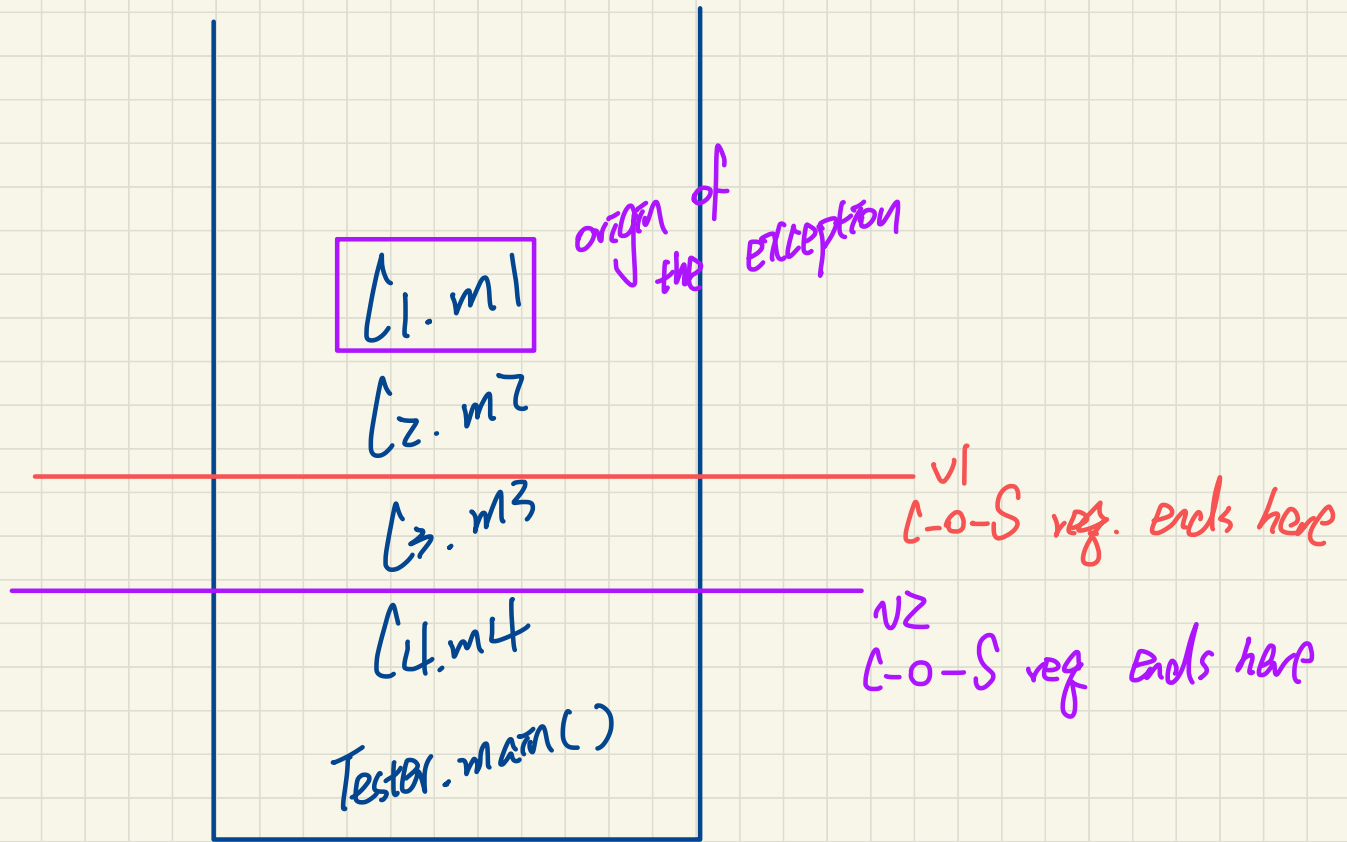
all callers underneath m are no longer subject to catch-or-specify req. anymore

entry point of exec.

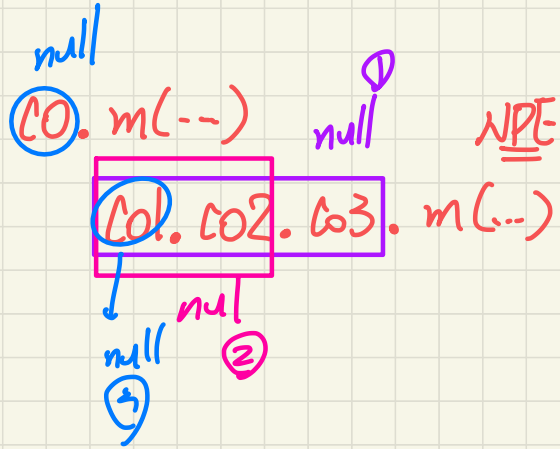
1st catch option (m)

calls between m and entry meth.

no longer subject to catch-or-specify req. anymore



NullPointerException



ArrayIndexOutOfBoundsException

$a[\underline{exp}]$

AOBE for: $0 \leq exp \leq a.length - 1$

Assertion Failure

assertTime (...);