

Exercise walk-through

4.9

ArrayList Traversals

AP Computer Science A

The ArrayList toolbox

An **ArrayList** grows and shrinks, unlike a fixed array. It stores objects, so numbers are **wrapped** (Integer, Double).

add(x)
append

get(i)
read

set(i,x)
replace

remove(i)
delete

Traverse it with a for or an enhanced for-each loop: `for (int x : list).`

Removing from a list while looping over it

the problem

`remove(i)` shifts every later element down, so the element after the removed one is skipped

fix one

loop backwards, from `size()-1` down to 0 – later shifts cannot affect indices you have passed

fix two

do not increment `i` after a removal, so the next pass re-examines the same index

never

remove inside an enhanced for loop – it throws a `ConcurrentModificationException`

the rest

the array algorithms carry over unchanged, plus insertion and deletion, which arrays cannot do easily

This is the single most-tested ArrayList point. If a question removes items in a loop, it is testing exactly this.

Exercises 2.1-2.3

2.1 State the index range for an indexed traversal of an ArrayList. [1]

2.2 Name the method that gives an ArrayList's number of elements. [1]

2.3 State one risk of removing elements while looping forward with an index. [1]

Exercises 2.1-2.3

2.1 State the index range for an indexed traversal of an ArrayList. [1]

2.1

- from 0 to `size() - 1`
-

2.2 Name the method that gives an ArrayList's number of elements. [1]

2.2

- `size()`
-

2.3 State one risk of removing elements while looping forward with an index. [1]

2.3

- the later elements shift down, so the next one can be skipped

Exercises 3.1-3.2

3.1 An indexed for loop over an ArrayList runs from 0 to

[1]

☐

A size

B size - 1

C length

D length - 1

3.2 To read each element of an ArrayList simply, use

[1]

☐

A a for-each loop

B a constructor

C a cast

D a static method

Exercises 3.1-3.2

3.1 An indexed for loop over an ArrayList runs from 0 to

[1]

☐

A size

B size - 1

C length

D length - 1

3.1 B

- size - 1.

3.2 To read each element of an ArrayList simply, use

[1]

☐

A a for-each loop

B a constructor

C a cast

D a static method

3.2 A

- a for-each loop.

Exercise 3.3 (a)

3.3 An `ArrayList` has 4 elements.

(a) State the index of the first element.

[1]

Exercise 3.3 (a)

3.3 An `ArrayList` has 4 elements.

(a) State the index of the first element.

[1]

(a)

- 0

Exercise 3.3 (b)

3.3 An `ArrayList` has 4 elements.

(b) State the index of the last element.

[1]

Exercise 3.3 (b)

3.3 An `ArrayList` has 4 elements.

(b) State the index of the last element.

[1]

(b)

- 3

Exercise 3.3 (c)

3.3 An `ArrayList` has 4 elements.

(c) Name the method giving the number of elements.

[1]

Exercise 3.3 (c)

3.3 An `ArrayList` has 4 elements.

(c) Name the method giving the number of elements.

[1]

(c)

- `size()`