Australian National University

R1

Review

“You can know the name of a bird in all the languages of the world, but when you're finished, you'll know absolutely nothing whatever about the bird... So let's look at the bird and see what it's doing -- that's what counts. I learned very early the difference between knowing the name of something and knowing something.”

Richard Feynman

Java

J1	Imperative programming, standard library, types	Q1
J2	Types, objects, classes, inheritance, interfaces	Q2
J3	Naming, literals, primitives	Q1,Q2
J4	Arrays, operators, expressions, statements, blocks	Q1,Q2
J5	if-then-else, switch	Q1,Q2
J6	while, do-while, for	Q1,Q2
J7	parameters, return values	Q1,Q2
J8	Nested classes	Q2

J10	Integer etc., autoboxing, Math, Random	
J11	Character and String	Q1
J12	Generics	Q1
J13	Type inference	Q1,Q2 Q3,Q4
J14	Collections and sorting	Q4
J15	Java exceptions, catch or specify, Java syntax	Q4
J16	Threads	Q5

Software Engineering

S1	IDEs, revision control, Gitlab and Git	Q2
S2	Revision Control	Q2
S3	Test Driven Development	Q1,Q2

Object Oriented Programming

O1	Class declaration, object creation	Q1,Q2
O2	Initializers, access control, nested classes, enum types	Q2
O3	Interfaces	Q2
O4	Inheritance, overriding, polymorphism, super	Q2,Q4
O5	java.lang.Object, final, abstract	

Java FX

JavaFX **not examinable** in main exam *(but may arise in supplementary exam in the context of questions about your assignment)*

Abstract Data Types (ADTs)

A1	List implementation 1	Q1,Q2,Q4,Q5
A2	List implementation 2	Q1,Q2,Q4,Q5
A3	The set ADT and its implementation	Q2,Q5
A4	Hash tables	
A5	The tree ADT	Q2
A6	Map ADT implementation, ADT recap	Q2

Core Computer Science

C1	Recursion	Q1
C2	Hash functions, choosing a good hash function	Q2
C3	Hashing applications, Java's hashCode()	
C4	Files	
C5	Computational Complexity	Q2,Q5
C6	Grammars	Q5
C7	Threads	Q5