Test

7 March 2001

Test

- Covering:
  - Chapters 1 – 10 (I/O Streams)
- Format:
  - 40 marks
    - aim for mark per minute
  - Short multiple choice section
  - Short answer questions

Questions

- Demonstrate your knowledge of the syntax and semantics of Java, and the important underlying concepts
- Be able to understand and to write methods and short code fragments
  - Not full programs
  - Possibly outline what needs to be in a class (design)

Discussion?

Questions?

Overriding Methods (cont.)

```
public class ChequeAccount extends Account
  { public double overdraftLimit;
    public double overdraftIntRate;

    public void withdraw (double amount)
      { ... // Allow for overdraft
      } // withdraw

  } // class ChequeAccount
```

The `Object` Class

- Superclass of *all* Java classes

```
public class X extends Object
  {
  } // class X
```

`Object` Methods

- *All* Java objects inherit some common methods from the **`Object`** class
  - **`toString`**
  - **`equals`**

The `toString` Method

- By default gives class name and memory address:

```
ChequeAccount cq = new ChequeAccount();
System.out.println(cq);
```

```
ChequeAccount@3fbdb0
```

- But, we can override it!

Overriding **toString**

```
public class ChequeAccount extends Account
  { ...

    public String toString ()
      { return (accountNumber + ” bal = R”
                + balance);
      } // toString

  } // class ChequeAccount
```

The `equals` Method

- Compares two objects
  - Default: compares the *references* (same as **==**)
- Can override
  - e.g. compare account numbers

Polymorphism

- *It's Greek to me!*
- Objects can take on different forms:

```
ChequeAccount c = new ChequeAccount();
Account a = new ChequeAccount();
```

?