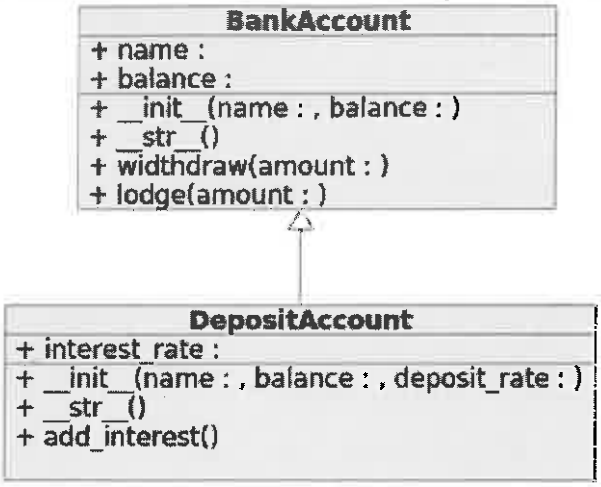
Q3 The following diagram illustrates an inheritance hierarchy for bank accounts:



The Python file `BankAccount.py` is provided in the `python` directory. The `BankAccount` class contains attributes (instance variables) representing the name of the account and the bank balance, an initialiser method `__init__`, a method `__str__` which returns a string representation of the account, and methods `withdraw` and `lodge` which allows withdrawals and lodgements to be made.

(a) In a separate program file, create a subclass of `BankAccount` called `DepositAccount` which represents a savings account, which includes:

(i) The class header. *(1 mark)*

(ii) An initialiser method `__init__` which takes three parameters (in addition to `self`): the first two parameters, `name` and `balance` should be passed to the superclass initialiser, and the remaining parameter `deposit_rate` should be used to initialise a corresponding attribute. *(4 marks)*

(iii) A method `__str__` which returns a string representation of an deposit account, including name, balance and deposit interest rate; use the superclass `__str__` method to provide the name and balance. Use the function `str()` to convert a number to a string. *(3 marks)*

(iv) A method `add_interest` which checks if the balance is greater than zero and if so, calculates the interest due using the formula:

$$interest = balance * \frac{rate}{100}$$

and adds it on to the balance using the `lodge` method from the superclass. *(4 marks)*

(b)Write a separate program which

- Imports the classes `BankAccount` and `DepositAccount`
- Creates a `BankAccount` object
- Prints the `BankAccount` object
- Creates a `DepositAccount` object
- Prints the `DepositAccount` object
- Calls `add_interest` on the `DepositAccount` object
- Prints the `DepositAccount` object again *(8 marks)*

(c) Test the program and take a screenshot of the output. *(5 marks)*

Upload the Python source code file and the output screenshot.

[25 marks]