### **Program (60 marks)**

The program should consist of the following classes (with no additional attributes or methods).

1) The Bike class has three attributes, which correspond to the bikeID, bikeDescription and the company’s name. The bikeID, bikeDescription and the company’s name are each represented as a string of text. The bikeID, bikeDescription is initialised in the constructor, by being assigned the value of the constructor's parameters. The Bike class has a constructor with two parameters which takes bikeDescription and company as parameters.The company name is assigned an empty string initially.

Each attribute has a corresponding accessor method and there is another method to set the company’s name to a new name. A display method will output (suitably annotated) the bikeID, bikeDescription and, if the company’s name is not an empty string, the company’s name too.

The CustomerDetails class has five attributes id, name, address, phone and e-mail. Student has to define proper data types to store data. Each attribute has a corresponding accessor method and mutator method except id attribute.. The id is being initialized in the constructor with only parameter.

**[10 marks]**

2)The BikeToRent class is a subclass of the Bike class and has seven  
attributes: Student has to define attributes with proper data-type to store data bikeHireDate, returnDate, numberOfDays, dailyRate, advancePayment, totalRentalCost thatbike has accumulated to-date and bike loanStatus to set either true orfalse.

The constructor accepts three parameters, which are the bikeDescription, companyName and the dailyrate. A call is made to the super class constructor with two parameter, the bikeDescription andcompany name. The daily rate attribute is assigned the corresponding parameter value. additionally,in the constructor, the other remaining attributes are assigned either with 0 or empty string("") andthe loan status is initialised to false.Each attribute has a corresponding accessor method.

A method is required to set the daily rate as changes to rates inevitably occur. The methodaccepts a new daily rate as a parameter and assigns the new value to the daily rate attribute.There is a method to rent out the bike to a customer. The method accepts, as parameters, anew customer's name, the date on which the bike is required, the date on which thebike is to be returned, the number of days the bike is required and the amount paidby the customer. If the bike is already on loan, an appropriate message including thedate of return is output to the customer. If the bike is available, the method to set thecustomer name is called with the customer's name as a parameter. The date on which thebike is required, the date on which the bike is to be return, the number ofdays the bike is required and the amount paid by the customer are all updated by theparameter values input to the method, the loan status of the device is changed to true, total rentalcharge which is equal to the daily rate \* the number of days, is added to the total rentaccumulated for that Hardware Device.

There should be a method for making the bike available to the Company. If thebike is not on loan, a suitable message is output. If the bike is on loan, the methodto set the customer's name is called with "" as a parameter, the number of days rental is set to 0,the date of hire and the date of return are each set to empty, total rental cost to 0 and the loanstatus is set to false.

There is a method to print the description of the bikeand its total rent collected to date, each suitably annotated. (To access the bikeDescription, company, the method to get thedescription must be called.)

A method to display the details of the bike is required. It must have the samesignature as the display method in the Bike class. It will call the method in Bike class to display the bike description, company and, if applicable, the customer'sname. The method should also then display the daily rate . If the bike is on loan thenthe date of hire, the date of return and the number of days must also be displayed. Each outputmust be suitably annotated.

**[20 marks]**

3) The **BikeToSell** class is also a subclass of the Bike class and it has five attributes. Student has to define attributes with proper data-type to store data price, tax amount, totalamount, selling date and bike selling status to set either true or false.

The constructor accepts four parameters which are the bikedescription, company, price and tax, acall is made to the super class constructor with two parameters, the bikeDescription and thecompany. The attributes price and tax are given the corresponding parameter values.Additionally, the constructor initializes sold to false and selling date to empty string.Each attribute has a corresponding accessor method.

A method is required to set the price as changes to price inevitably occur. The method accepts anew price as a parameter and, if the bike has not been sold, the new value isassigned to the price attribute. If the bike has already been sold then a suitablemessage is output to the user indicating that it is therefore not possible to change the price.Another method is also required to set the tax rate as changes to tax inevitably occur. Themethod accepts a new tax rate as a parameter and, if the bike has not been sold, thenew value is assigned to the tax attribute. If the bike has already been sold then asuitable message is output to the user indicating that it is therefore not possible to change the tax  
rate.

There is a method for selling bike. The method has two parameters, the customer'sname and selling date. If the bike has not been sold, the method to set thecustomer's name is called with the customer's name as a parameter, set selling date with theparameter value, set the total cost for the particular bike and sold is set to true, otherwise asuitable message is output to the user indicating that the Hardware device has already been sold.

A method to display the details of the bike is required. It must have the same  
signature as the display method in the Bike class. It will call the method in  
Bike class to display the description and, if applicable, the customer name. If the  
bike has not been sold the price must also be displayed. Each output must be  
suitably annotated.