

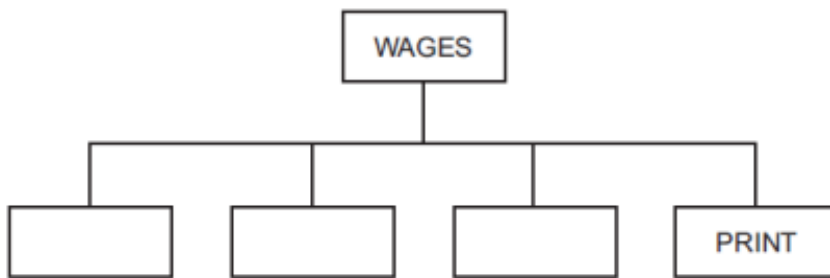


## 2.1.2 Structure Chart

May/June 2011. P23

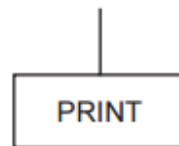
3. Kris has written a program that will work out the wages for her staff. The main steps for each employee are: to work out the hours worked, work out the total earnings, work out tax and finally print out how much will be taken home.

(a) Complete the structure diagram to show the modules that will be needed.



[3]

(b) The printout will be different for those staff who receive cash and those who have their earnings paid directly to a bank account. Add the next level to the print module.



[2]

Oct/NOV 2011 P21

2 Nathan is designing a software solution for stock control in a mobile phone shop. He has a colleague, called Andre, who will help him write the program. Nathan decides to modularise the solution.

(a) State why modularisation is a good idea.

[1]

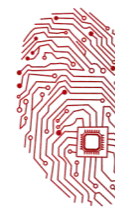
(b) As the first step in his design he splits the solution into the following main areas:

Initialisation, PhoneSales, StockOrdering, Accounts.

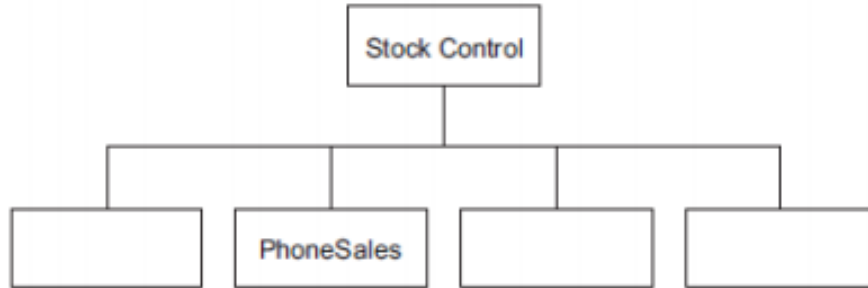
Complete the following structure diagram.

[1]





### 2.1.2 Structure Chart

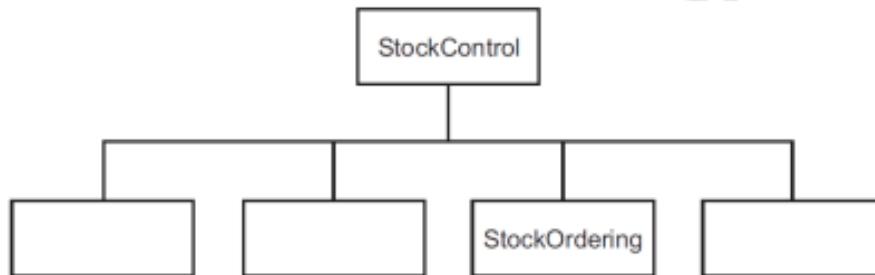


- (c) PhoneSales is made up of two modules, ShopSales and OnlineSales. Add them to the structure diagram shown in (b). [2]

#### Oct/NOV 2011 P22

1 Nathan is designing a software solution for stock control in a mobile phone shop. He has a colleague called Andre who will help him write the program. Nathan decides to modularize the solution.

- (a) State why modularisation is a good idea. [1]  
(b) As the first step in his design he splits the solution into the following main areas: Initialisation, PhoneSales, StockOrdering, Accounts. Complete the following structure diagram.



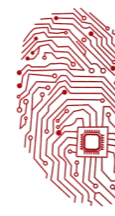
- (c) StockOrdering is made up of two modules, PlaceOrder and ReceiveGoods. Add them to the structure diagram in (b). [2]

#### Oct/NOV 2011.P23

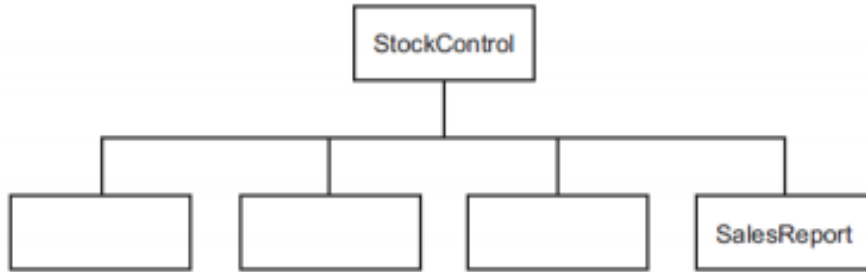
1 Nathan is designing a software solution for stock control in a computer shop. He has a colleague, called Andre, who will help him write the program. Nathan decides to modularise the solution.

- (a) State why modularisation is a good idea. [1]  
(b) As the first step in his design he splits the solution into the following main areas: Initialisation, StockOrdering, Sales, SalesReport. Complete the following structure diagram.





### 2.1.2 Structure Chart



[1]

- (c) SalesReport is made up of two modules, MonthlySalesReport and AnnualSalesReport. Add them to the structure diagram in (b).

[2]

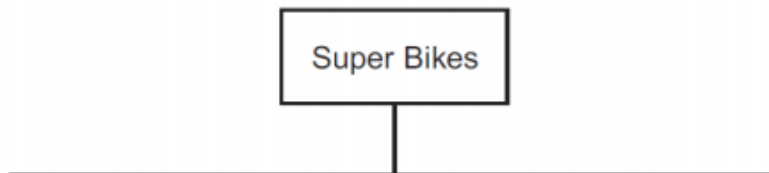
Oct/NOV 2012.P21

1 Soni works for a software house which has been asked to design software for a cycle hire company, Super Bikes.

Soni decides on the main tasks:

- collecting the information about new bikes
- entering details of repairs
- entering details of hirer
- entering details of payment

(a) Complete the structure diagram showing these tasks.



[2]

(b) The collection of information about repairs has three subtasks:

- input the repair needed
- input the parts list
- input the cost of the repair

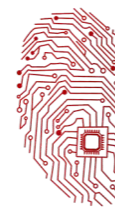
Add these to the structure diagram in part (a).

[1]

(c) State two reasons for dividing the main task into smaller tasks.

[2]





## 2.1.2 Structure Chart

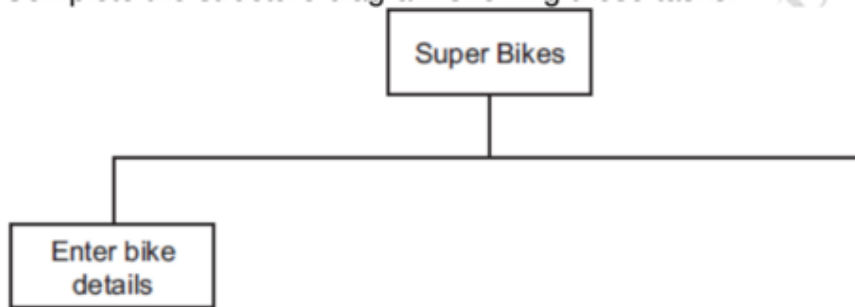
### Oct/NOV 2012 P22

Soni works for a software house which has been asked to design software for a car hire company, Super Cars.

1 Soni decides on the main tasks:

- enter car details
- enter car hire details for
  - hirer
  - car
  - payment

(a) Complete the structure diagram showing these tasks.



(b) State two reasons for dividing the main task into smaller tasks. [2]

(c) Entering car details has two subtasks: [2]

- input the car's identification details
- input the hire rates for that car

Add these to the structure diagram in part (a). [1]

### Oct/NOV 2012 P23

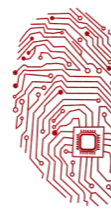
Soni works for a software house which has been asked to design software for Super Bikes, a company that specialises in hiring out motorbikes.

1 Soni decides on the main tasks:

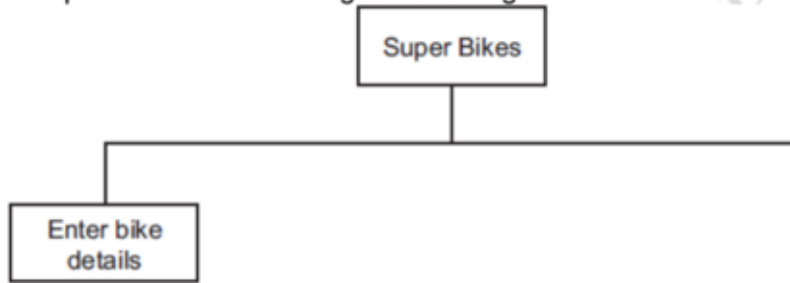
- enter bike details
  - bike specification
  - service history
  - hire rates
- enter bike hire details

(a) Complete the structure diagram showing these tasks.





### 2.1.2 Structure Chart



[2]

(b) Entering hire rate details has two subtasks:

- input daily rates
- input weekly rates

Add these to the structure diagram in part (a).

[1]

Oct/Nov 2013. P21

1 The Computing Department has a problem keeping track of its teaching resources. Juan, a student, has been asked to design and program a solution as his computing project. It will be the first large problem he has worked on.

He intends to write one large program that follows the process right through. His teacher tells him to break the problem into smaller parts.

(a) State and justify three of the reasons his teacher could give him for breaking the problem into smaller parts.

[6]

Juan decides that the design will include the following modules:

- update the resource file when a new teaching resource is purchased
  - input all the data about the resource
  - generate a resource ID for the resource
  - store in the resource file
- update the resource file when a current resource is discarded

(b) (i) Describe one diagrammatic method for showing how these modules are related.

[2]

(ii) Use your method with the modules above.

[2]

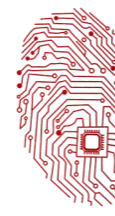
(c) Name two features of a high-level programming language that help with this modular approach.

[2]

(d) Juan realises that he will have to pass data from one module to another. Explain how this is done.

[2]





## 2.1.2 Structure Chart

### Oct/Nov 2013. P22

1 Jemma is designing a program that will work out the end of year bonuses for her employees. The main steps are:

- input employee's data
- calculate the bonus
- calculate deductions o tax o optional contribution to charity
- print out the bonus

(a) Draw a structure diagram to show the modules that will be needed.

[3]

### Oct/Nov 2013. P23

1 The Science Department has a problem keeping track of its equipment. Ashvin has been asked to design and program a solution as his Computing Project.

(b) This project was the largest computing problem Ashvin had worked on.

His friends wrongly advised him to program a solution as a whole.

He decided to break the problem into smaller parts. State and justify three of the reasons Ashvin gave his friends for breaking the problem into smaller parts.

[6]

Ashvin decides the design will include the following modules:

- Update the equipment file when new equipment is bought.
  - Generate an equipment ID for an item of science equipment.
  - Input all the data about the piece of equipment.
- Update the equipment file when old equipment is discarded.

(c) (i) Describe how a structure diagram will help Ashvin.

[2]

(ii) Draw a structure diagram for the above modules.

[2]

(d) Ashvin splits the first alternative module (Generate an equipment ID for an item of science equipment) into two smaller modules:

- Generate an equipment ID for an item of science equipment.
- Store the item record in the resource file.

Re-draw your diagram adding the two new modules.

[2]

(e) The module that stores all the equipment data needs to receive a data value produced in another module.

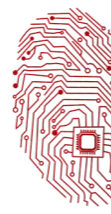
(i) What data value is this?

[1]

(ii) Explain how this is done.

[2]





## 2.1.2 Structure Chart

### Computer Science (9608)

May/June 2015. P21/P22

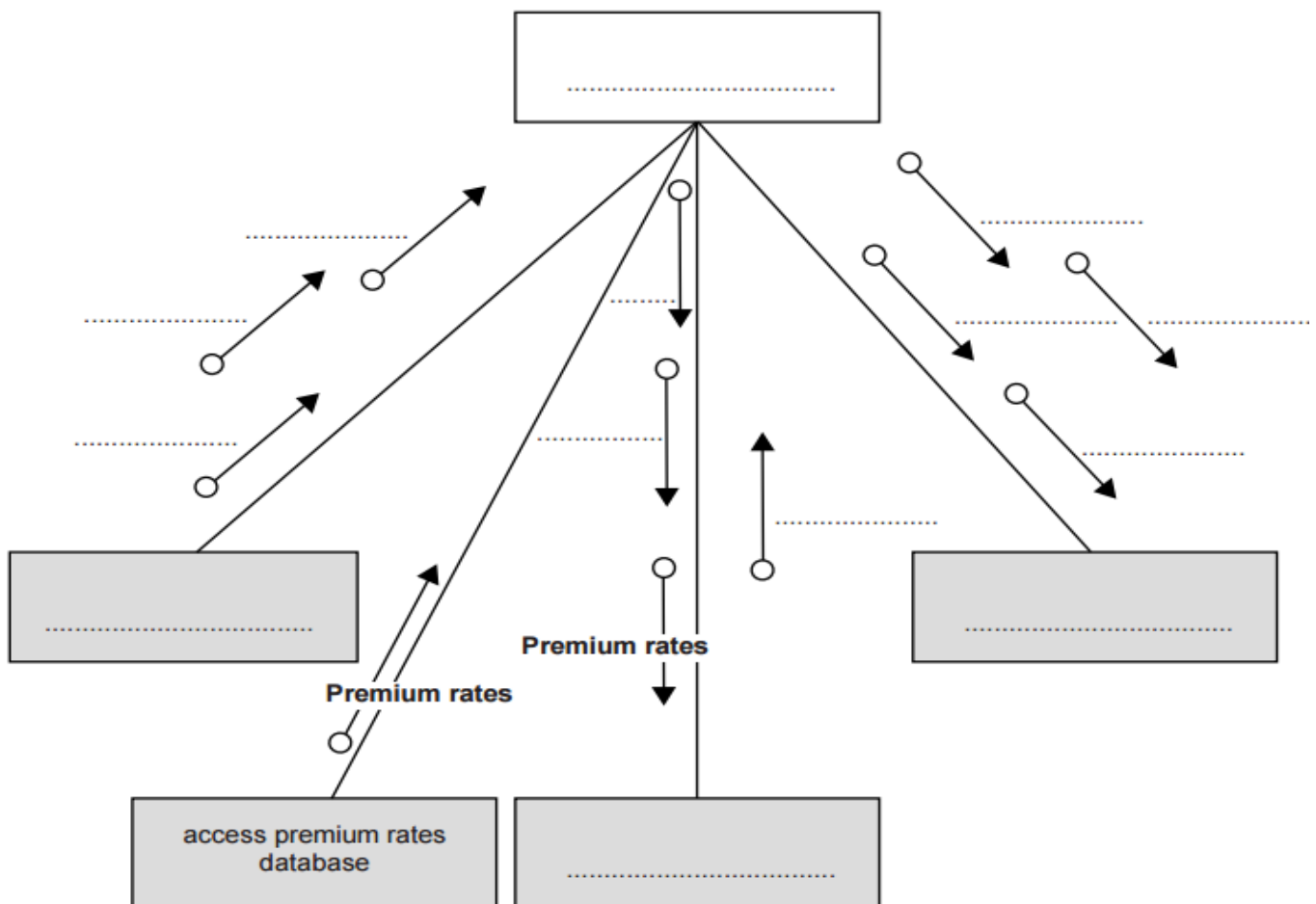
3 When the guarantee on a computer runs out, the owner can take out insurance to cover breakdown and repairs. The price of the insurance is calculated from:

- the model of the computer
- the age of the computer
- the current insurance rates

Following an enquiry to the insurance company, the customer receives a quotation letter with the price of the insurance.

A program is to be produced.

The structure chart below shows the modular design for this process:





### 2.1.2 Structure Chart

(a) Using the letters **A** to **D**, add the labelling to the chart boxes on the opposite page.

Modules	
A	Send quotation letter
B	Calculate Price
C	Produce insurance quotation
D	Input computer details

[2]

(b) Using the letters E to J, complete the labelling on the chart opposite. Some of these letters will be used more than once.

Modules	
E	CustomerName
F	CustomerEmail
G	Model
H	Age
I	PolicyCharge
J	PolicyNumber

[4]

#### May/June 2015. P23

3 A flooring company provides for each customer an estimated price for a new job. Each job is given a Job ID.

The job cost is calculated from the length (nearest metre) and width (nearest metre) of the room.

The process for calculating the price is as follows:

- the floor area is calculated with 18% added to allow for wastage
- the job cost is calculated at \$50 per square metre

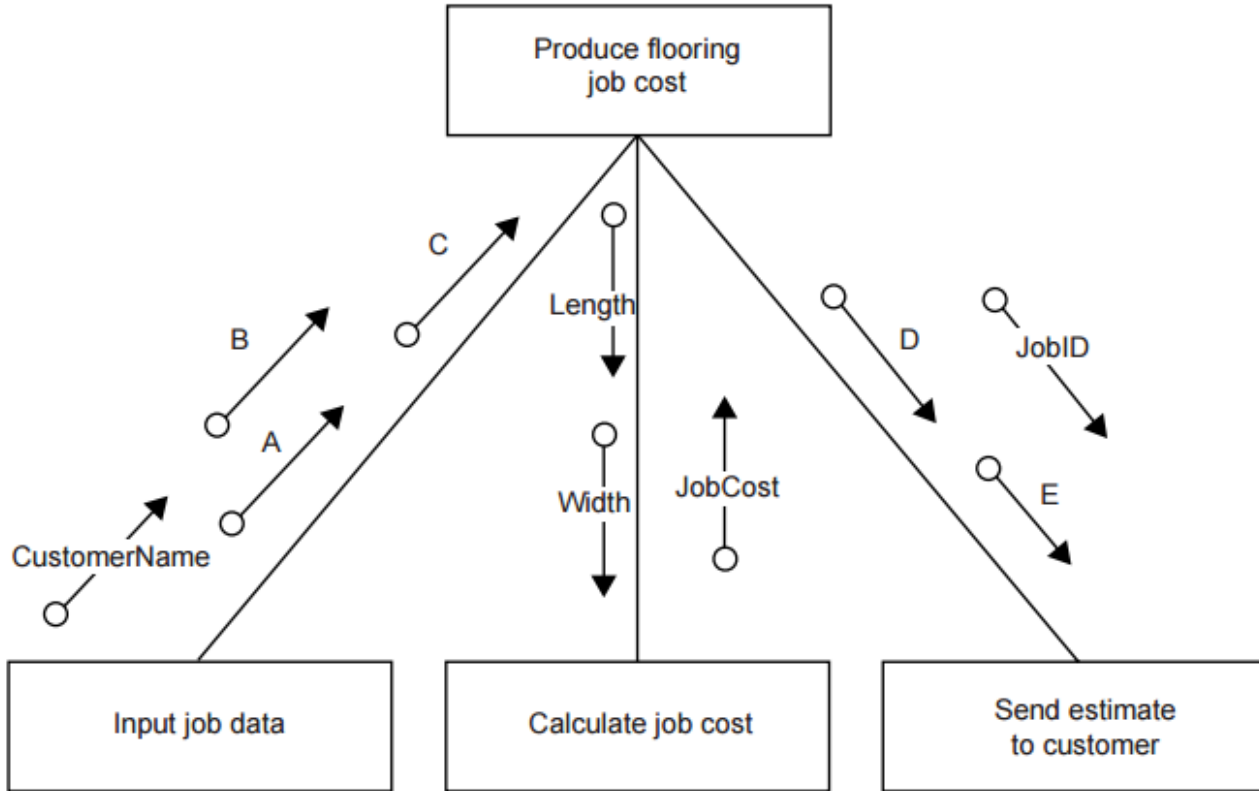
The structure chart shows the modular design for a program to produce a new job cost







### 2.1.2 Structure Chart



(i) Give the data items corresponding to the labels A to E in the structure chart.

[5]

May/ June 2016. P21/P22

**4 (a)** Structured programming involves the breaking down of a problem into modules.

Give **two** reasons why this is done.

[2]

**(b)** A team needs to write a program to implement an online shopping system. Customers will access the program via a website.

Customers can search for items before adding them to a virtual shopping basket. When they have finished shopping, they pay for the items. The program provides output for the dispatch of the items.

Some of the key features of the system are as follows:

- a customer can add many items to the shopping basket
- payment may be either by credit or debit card, or by adding to a customer account
- the shop may dispatch the items in one or more packages

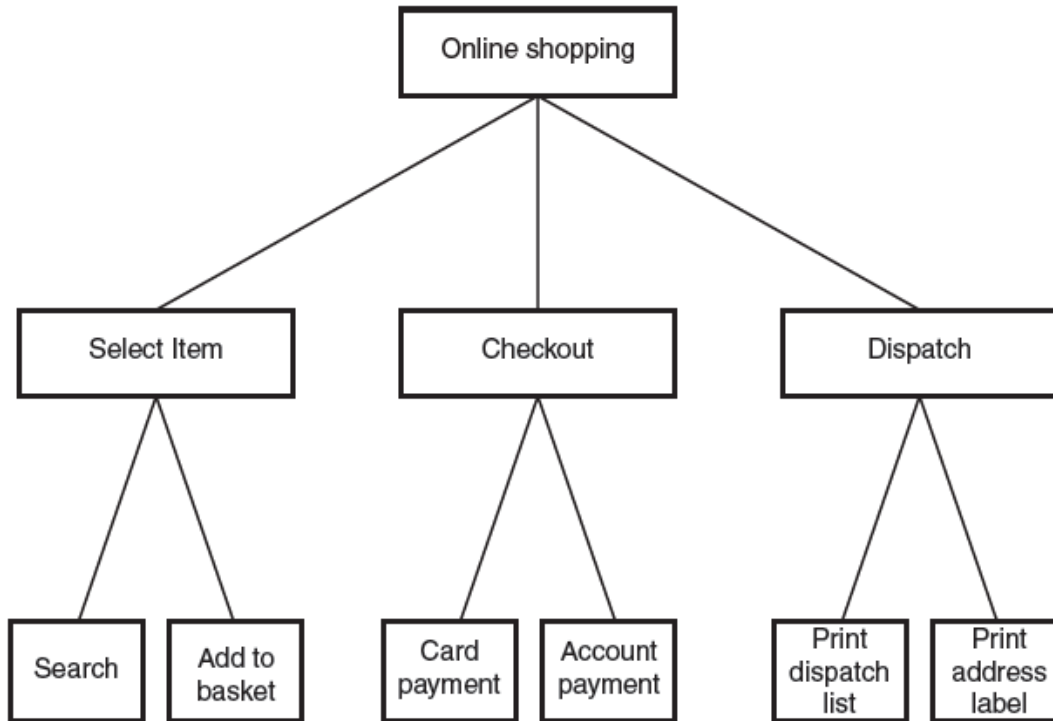
The structure chart below shows the program modules only.

(i) Draw on the chart, the symbols that represent the key features listed in **part (b)** above.



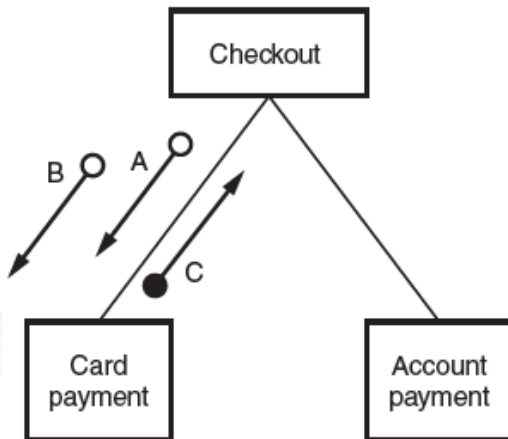


2.1.2 Structure Chart



[3]

(ii) A section of the chart in part (b)(i) is shown below. It is to show the parameters passed between the Checkout and Card payment modules.

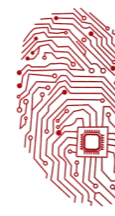


Name the three data items corresponding to the arrows.

Arrow	Data item
A	
B	
C	

[3]



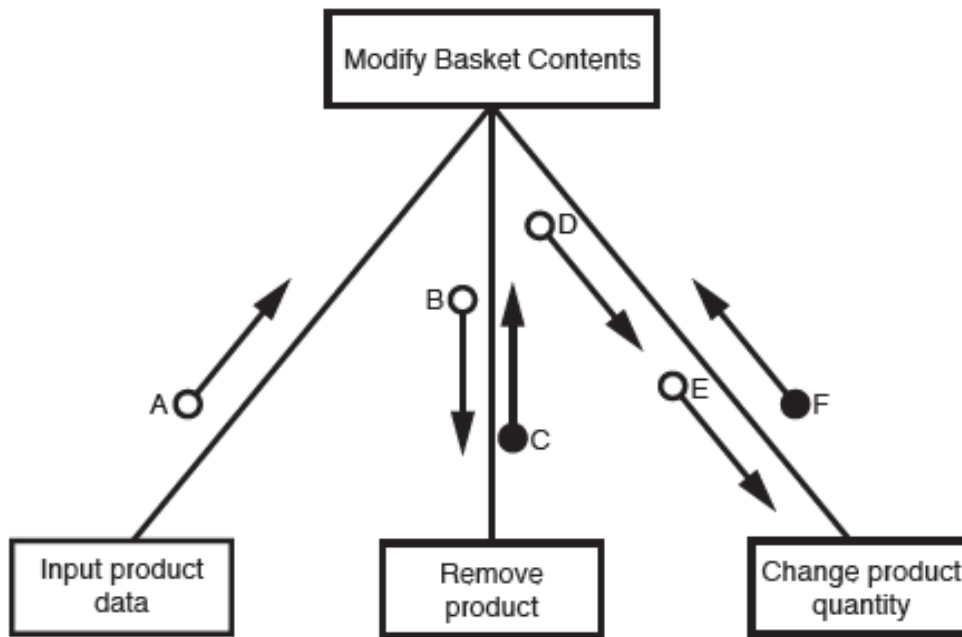


### 2.1.2 Structure Chart

May/ June 2016. P23

4 (a) Name **two** features of your chosen high-level programming language that support the implementation of a modular design. [2]

(b) (i) The structure chart shows part of the design of a program for an online shopping system. The user has already added a number of products to their virtual basket. Draw on the chart, the symbol to show that the process of modifying the basket contents may be iterated (repeated).



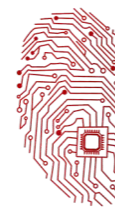
[1]

(ii) Each arrow in the structure chart above represents a parameter. The table below shows the three data items that the six parameters pass between modules. Tick (✓) to match each parameter to the correct data item.

Data item	Parameter					
	A	B	C	D	E	F
Product ID						
Quantity						
Flag Value – indicating operation success or fail						

[4]





## 2.1.2 Structure Chart

May/June 2018. P21

**4 (a)** A structure chart is used in modular program design.

Iteration and selection are two features of an algorithm that may be shown on a structure chart.

Give **three** other features.

**(b)** Pseudocode for a function is shown.

```
FUNCTION ItemProcess (AddItem, InString : STRING) RETURNS BOOLEAN
  DECLARE RetFlag : BOOLEAN
  RetFlag ← FALSE

  IF AddItem = "Yes"
    THEN
      RetFlag ← AddToList(InString)
    ELSE
      CALL RemoveFromList(InString)
  ENDIF

  RETURN RetFlag

ENDFUNCTION
```

Draw a structure chart on the next page to represent this pseudocode.

[8]

