12.2. State-transition diagrams and Structure charts: (Topical Past papers 2015 – 2024)

9608/41/M/J/15

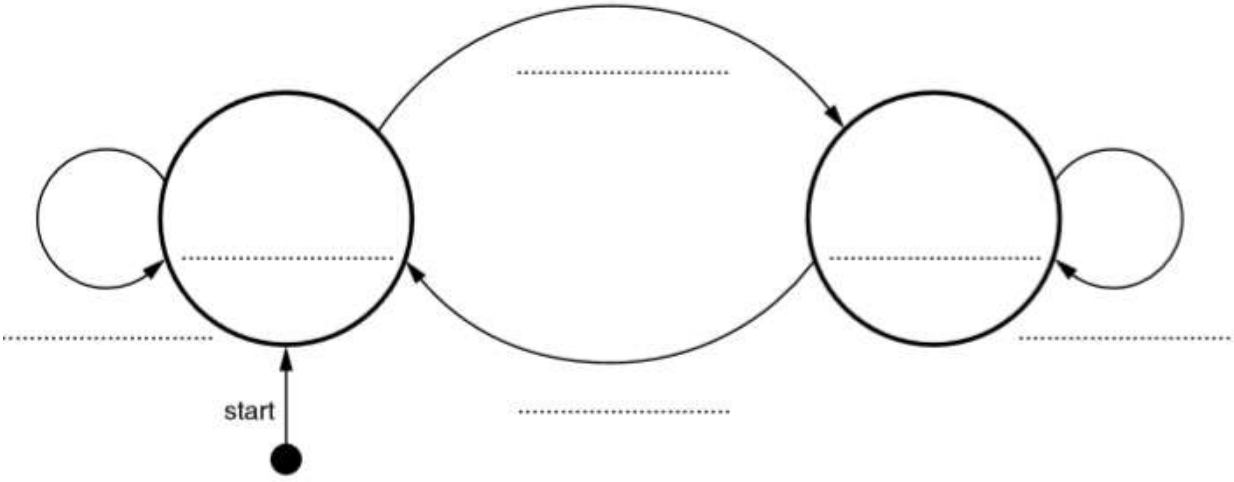
1/- A turnstile is a gate which is in a locked state. To open it and pass through, a customer inserts a coin into a slot on the turnstile. The turnstile then unlocks and allows the customer to push the turnstile and pass through the gate.

After the customer has passed through, the turnstile locks again. If a customer pushes the turnstile while it is in the locked state, it will remain locked until another coin is inserted.

The turnstile has two possible states: **locked** and **unlocked**. The transition from one state to another is as shown in the table below.

Current state	Event	Next state
Locked	Insert coin	Unlocked
Locked	Push	Locked
Unlocked	Attempt to insert coin	Unlocked
Unlocked	Pass through	Locked

Complete the state transition diagram for the turnstile:



[5]

(9608/43/M/J/15)

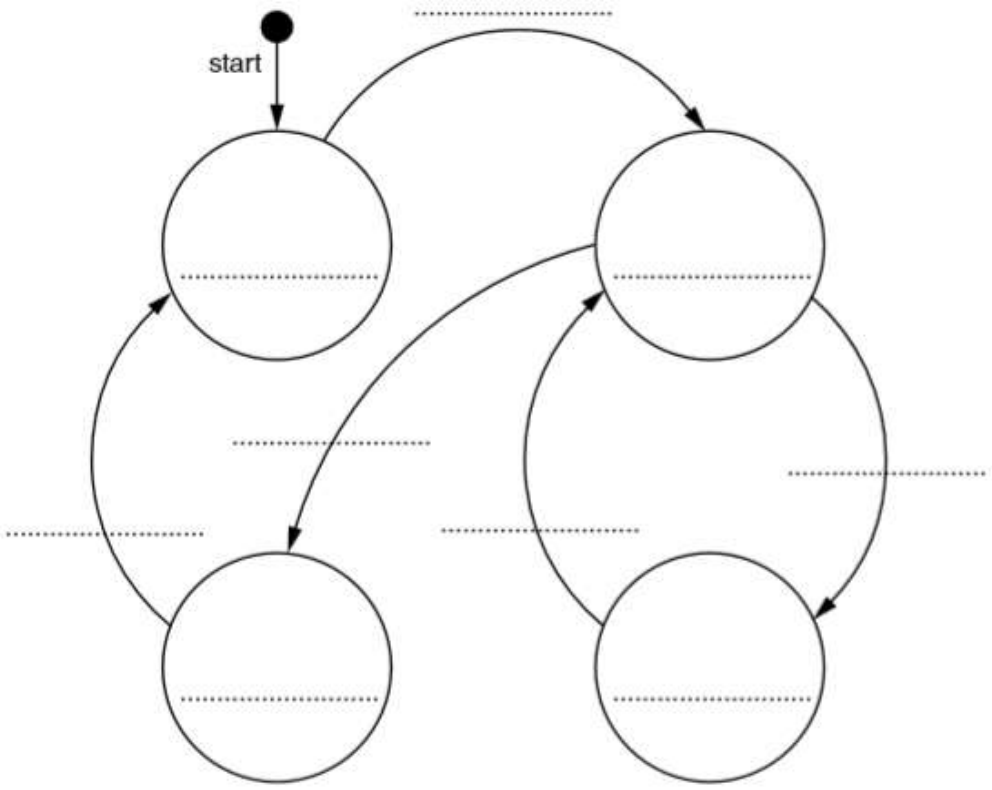
Q2/ - A petrol filling station has a single self-service petrol pump.

A customer can use the petrol pump when it is ready to dispense petrol. The pump is in use when the customer takes the nozzle from a holster on the pump. The pump dispenses petrol while the customer presses the trigger on the nozzle. When the customer replaces the nozzle into the holster, the pump is out of use. The cashier must press a reset button to make the pump ready for the next customer to use.

The petrol pump's four possible states and the transition from one state to another are as shown in the table below.

Current state	Event	Next state
Pump ready	Take nozzle	Pump in use
Pump in use	Press trigger	Pump dispensing
Pump dispensing	Stop pressing trigger	Pump in use
Pump in use	Replace nozzle	Pump out of use
Pump out of use	Reset pump display	Pump ready

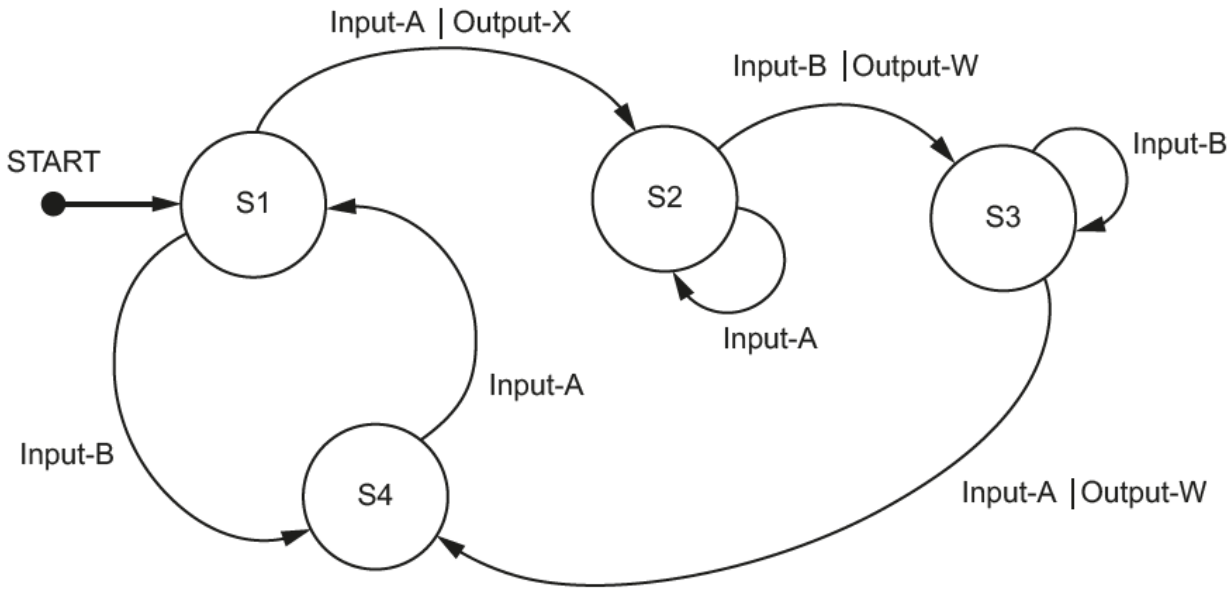
Complete the state transition diagram for the petrol pump:



[9]

9618/22/O/N/22

3. Examine the following state-transition diagram.



Complete the table to show the inputs, outputs and next states.

Input	Output	Next state
		S1
Input-A		
		S2
	Output-W	
	Output-W	

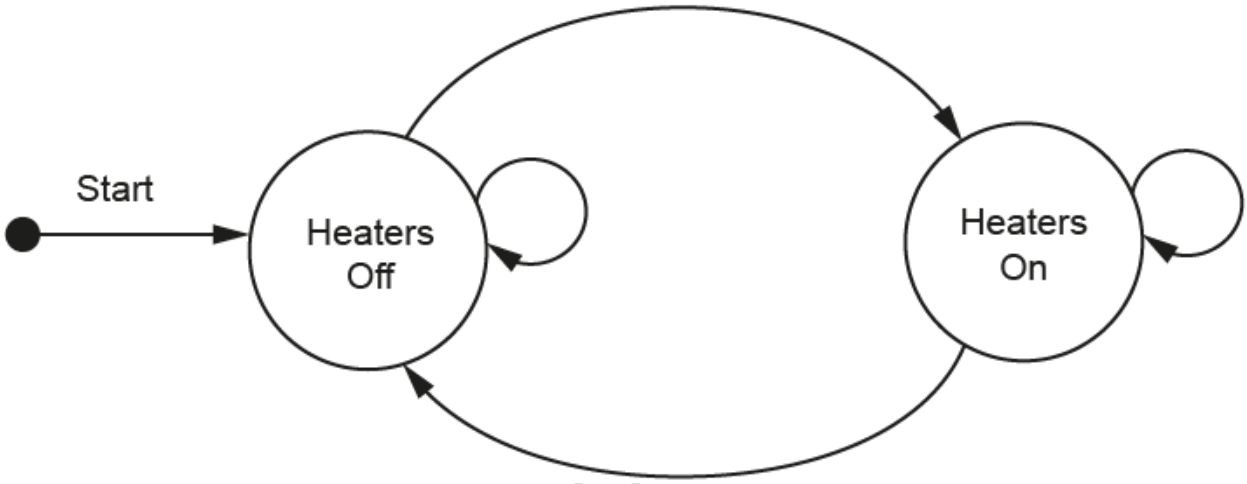
[4]

9618/21/O/N/21

4. A program module controls the heaters. This module operates as follows:

- If the temperature is below 10, switch the heaters on.
- If the temperature is above 20, switch the heaters off.

Complete the following state-transition diagram for the heating system:

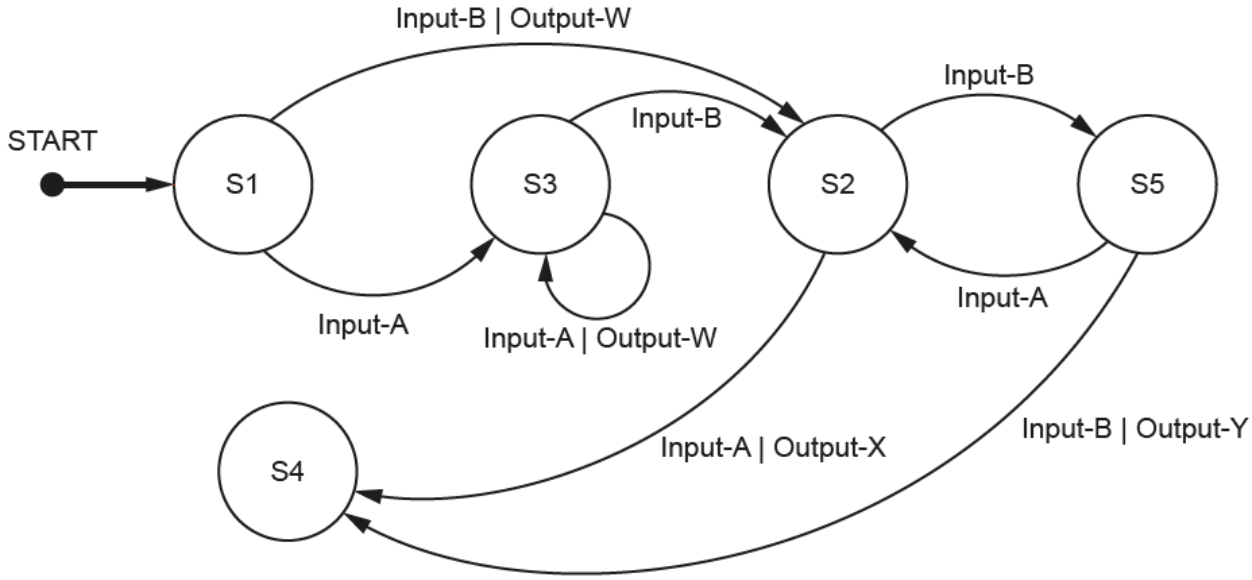


[3]

9618/22/M/J/24

5.

A different part of the program is represented by the following state-transition diagram.



(i) Complete the table to show the inputs, outputs and next states.
Assume that the current state for each row is given by the 'Next state' on the previous row.

For example, the first Input-A is made when in state S1.

If there is no output for a given transition, then the output cell should contain **'none'.**
The first two rows have been completed.

Input	Output	Next state
		S1
Input-A	none	S3
	Output-W	
	none	
Input-B		
Input-A		
		S4

[5]

(ii) Identify the input sequence that will cause the minimum number of state changes in the transition from S1 to S4.

..... [1]

9618/22/M/J/23

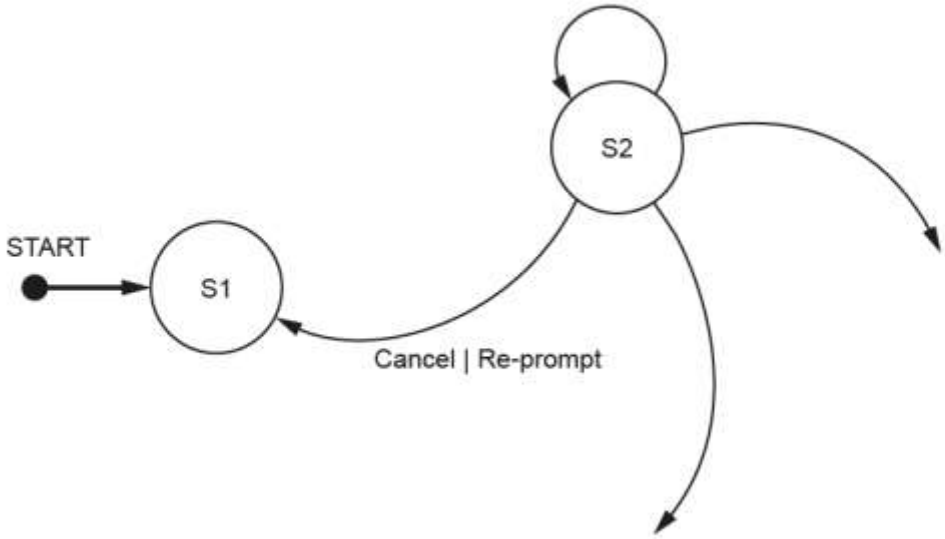
6. The program includes a module to validate a Personal Identification Number (PIN). This is used when customers pay for goods using a bank card.

A state-transition diagram has been produced for this module.

The table show the inputs, outputs and states for this part of the program:

Current state	Input	Output	Next state
S1	Input PIN		S2
S2	Re-input PIN	Display error	S2
S2	Cancel	Re-prompt	S1
S2	Valid PIN	Enable payment	S4
S2	Too many tries	Block Account	S3

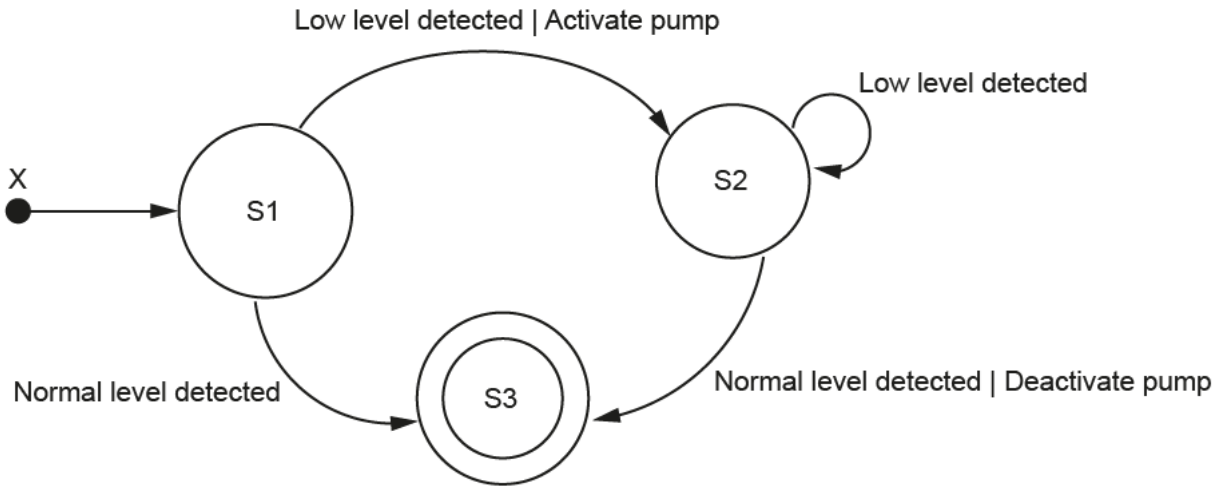
Complete the state-transition diagram to represent the information given in the table.



[4]

9618/22/M/J/21

7. Examine the following state-transition diagram



(i) Complete the table with reference to the diagram.

	Answer
The number of transitions that result in a different state	
The number of transitions with associated outputs	
The label that should replace 'X'	
The final or halting state	

[4]

(ii) The current state is S1. The following inputs occur.

1. Low level detected
2. Low level detected
3. Low level detected
4. Low level detected

Give the number of outputs and the current state.

Number of outputs .....

Current state ..... [2]

9618/21/O/N/24

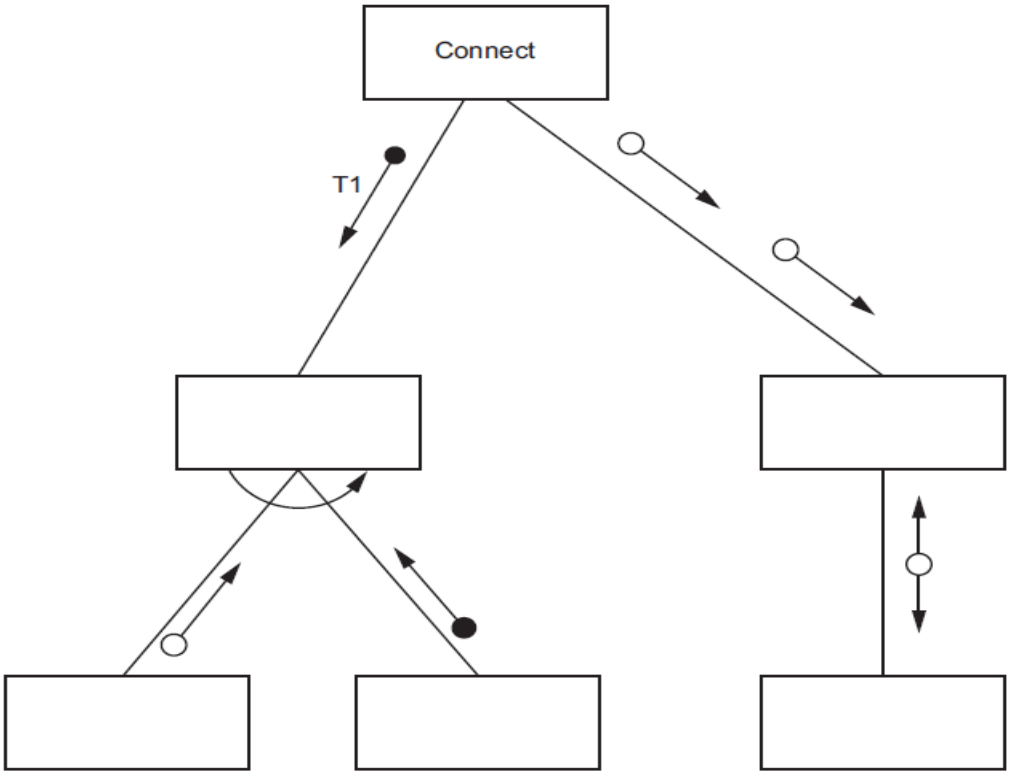
8. A program contains six modules with headers as follows:

Pseudocode module header
`PROCEDURE Connect()`
`FUNCTION Activate(H1 : STRING, Code : INTEGER) RETURNS BOOLEAN`
`FUNCTION Sync(T1 : BOOLEAN, S2 : REAL) RETURNS INTEGER`
`PROCEDURE Initialise(BYREF ID : INTEGER, BYVAL CC : INTEGER)`
`FUNCTION Reset(RA : STRING) RETURNS INTEGER`
`FUNCTION Enable(SA : INTEGER) RETURNS BOOLEAN`

Module `Connect()` will call either `Activate()` or `Sync()`. This is decided at run-time.

(a) Complete the structure chart for these modules.

Majid Tahir at www.majidtahir.com



[5]

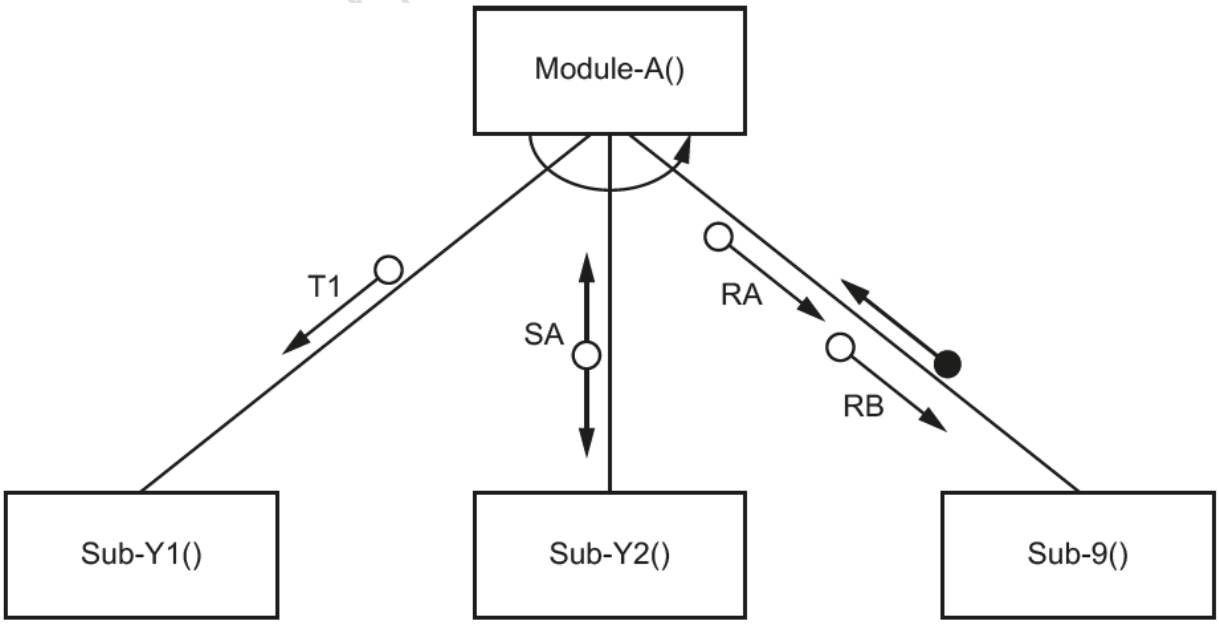
(b) Explain the meaning of the curved arrow symbol used in the diagram in part (a).

.....

..... [2]

9618/23/O/N/23

9. A structure chart shows the modular structure of a program:



(a) Explain the meaning of the curved arrow symbol which begins and ends at `Module-A ().`

.....

.....

.....

..... [2]

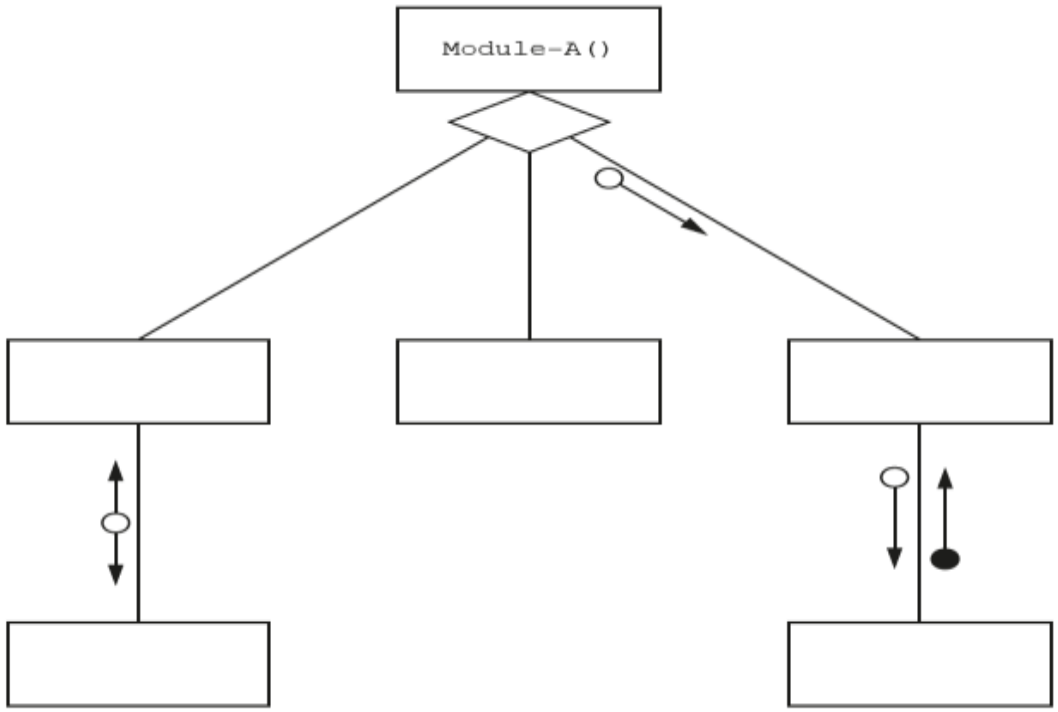
9618/22/O/N/23

10. A program contains six modules:

Pseudocode module header
`PROCEDURE Module-A()`
`PROCEDURE Module-X(T1 : INTEGER, S2 : REAL)`
`PROCEDURE Reset(BYREF Code : INTEGER)`
`FUNCTION Restore(OldCode : INTEGER) RETURNS BOOLEAN`
`FUNCTION Module-Y(RA : INTEGER, RB : BOOLEAN) RETURNS BOOLEAN`
`FUNCTION Module-Z(SA : INTEGER) RETURNS INTEGER`

```
Module-X() calls Reset()
Module-Y() calls Restore()
```

(a) Complete the structure chart for these modules.



[4]

(b) Explain the meaning of the diamond symbol as used in the diagram in part **(a)**.

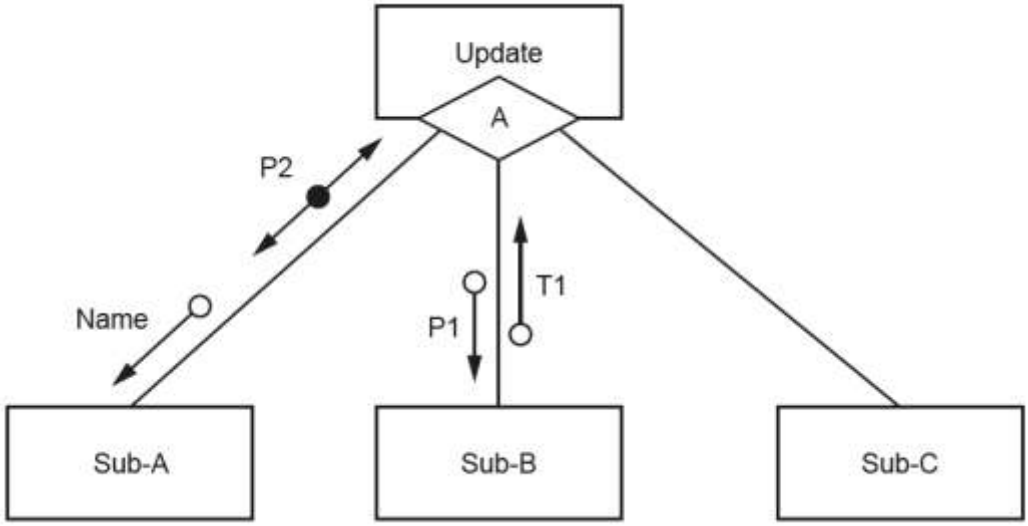
.....

.....

.....

..... [2]

11 . The structure chart illustrates part of the membership program:



Data item notes:
- Name contains the name of a club member
- P1 and T1 are of type real.

(i) Explain the meaning of the diamond symbol (labelled with the letter A) in the chart.

.....

.....

.....

..... [2]

(ii) Write the pseudocode module headers for Sub-A and Sub-B.

Sub-A

.....

.....

Sub-B

.....

..... [4]

Topical past papers by Majid Tahir

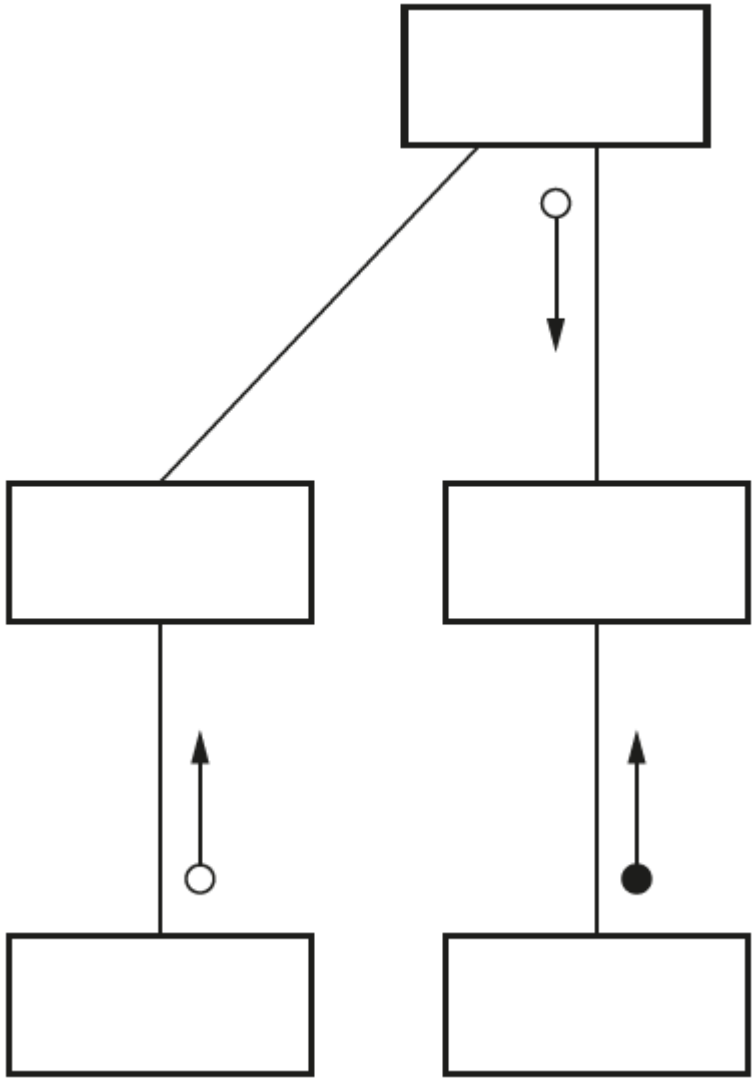
9618/23/M/J/23

12. Part of the library program contains program modules with headers as follows:

Pseudocode module header
`PROCEDURE Module-X()`
`PROCEDURE Module-Y(BYREF RA : INTEGER, SA : REAL)`
`PROCEDURE Overlay()`
`FUNCTION Replace(RA : INTEGER, RB : BOOLEAN) RETURNS BOOLEAN`
`FUNCTION Reset(TA : STRING) RETURNS INTEGER`

`Module-X()` and `Module-Y()` are both called from module `Overlay()`.

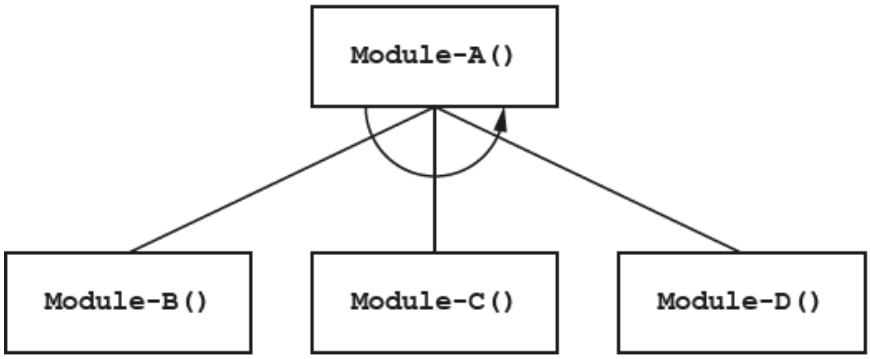
Complete the structure chart.



[3]

9618/21/M/J/23

13 The program designer produces a structure chart for the new module. Part of the structure chart is shown:



(i) Explain the relationship between the four modules shown.

.....

.....

.....

..... [2]

(ii) Two new modules are added: `Module-X()` and `Module-Y()`.

- `Module-X()` has no parameters.
- `Module-Y()` will take a string and a real number as parameters and return a Boolean value.
- `Module-D()` will call either `Module-X()` or `Module-Y()`.

Draw **only** the part of the structure chart that represents the relationship between `Module-X()`, `Module-Y()` and `Module-D()`.

[3]

www.Majid Tahir at www.majidtahir.com

9608 /MJ 2015/ 21

14. When the guarantee on a computer runs out, the owner can take out insurance to cover breakdown and repairs.

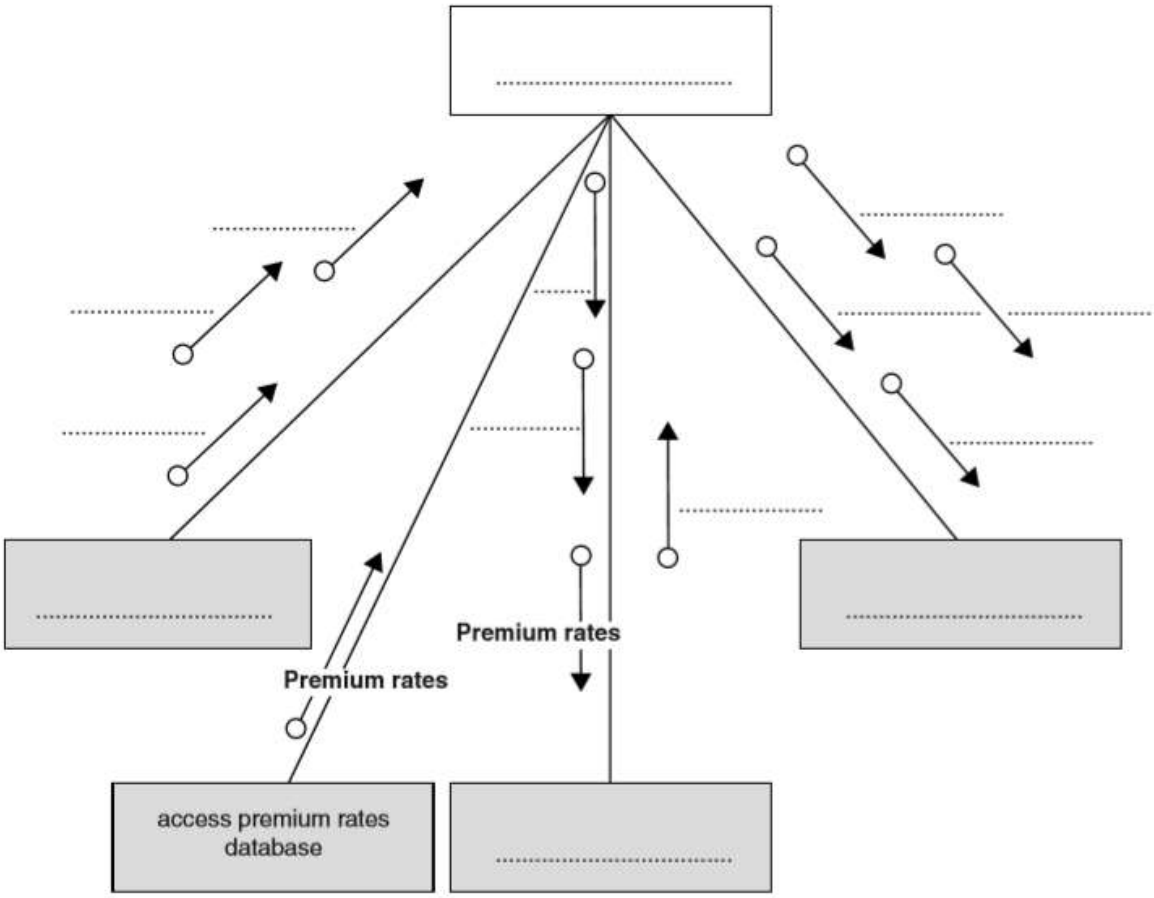
The price of the insurance is calculated from:

- 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:



(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.

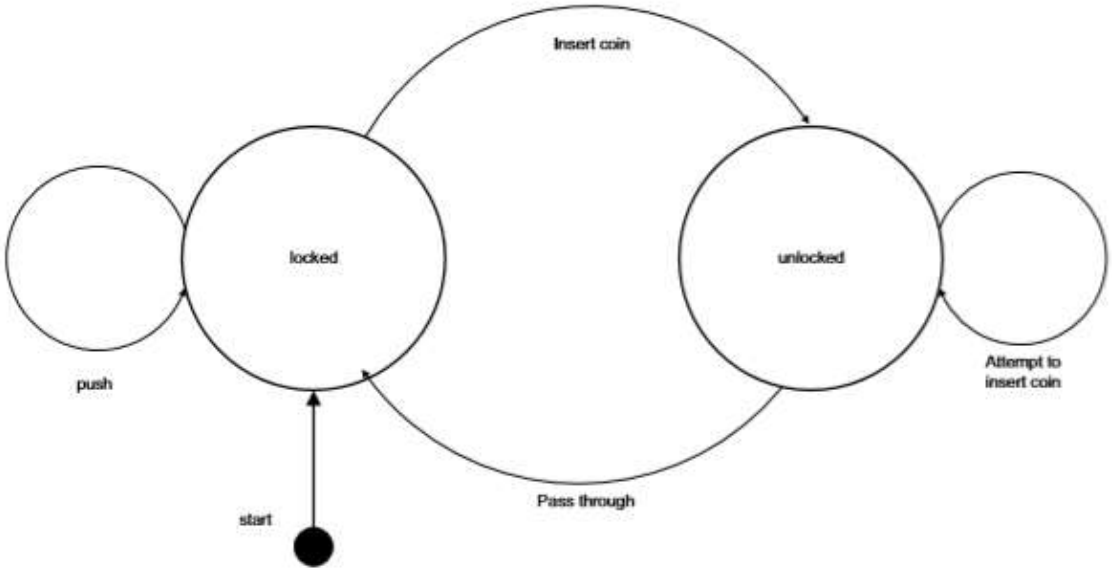
Data items	
E	CustomerName
F	CustomerEmail
G	Model
H	Age
I	PolicyCharge
J	PolicyNumber

[4]

Answers

9608/41/M/J/15

1

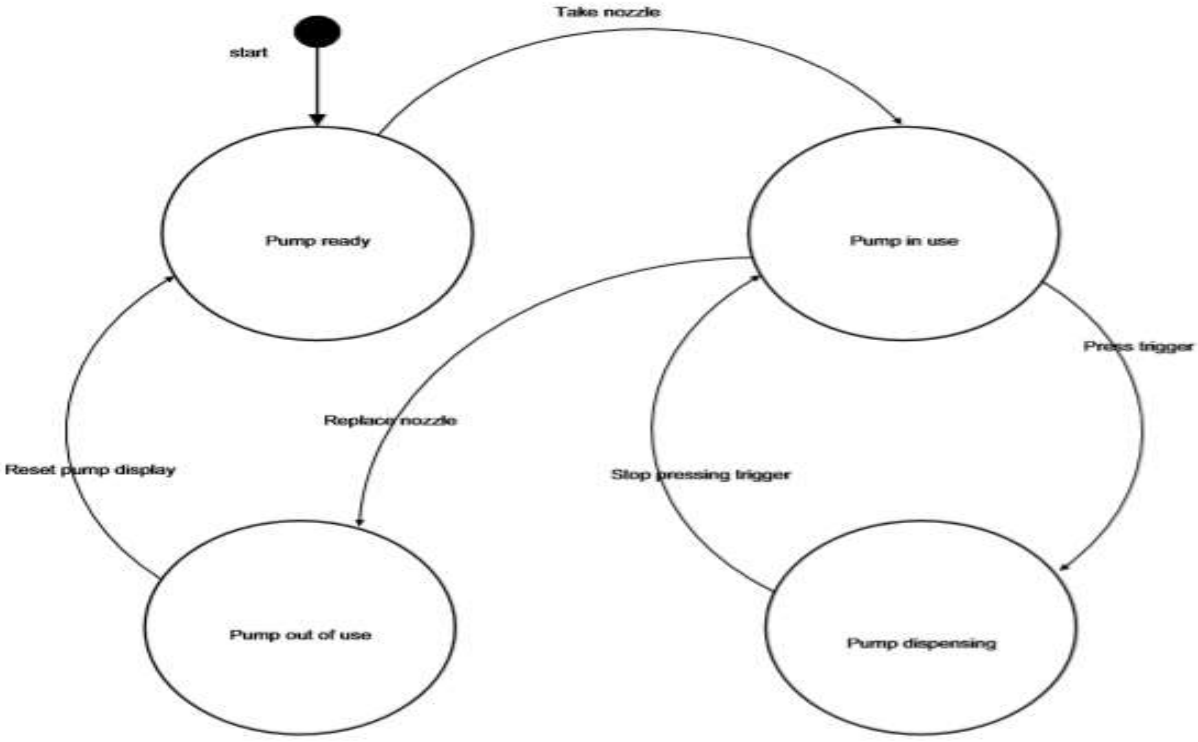


Mark as follows:
1 mark for both states correct
1 mark for each further label

[5]

(9608/43/M/J/15)

2



[9]

Q3

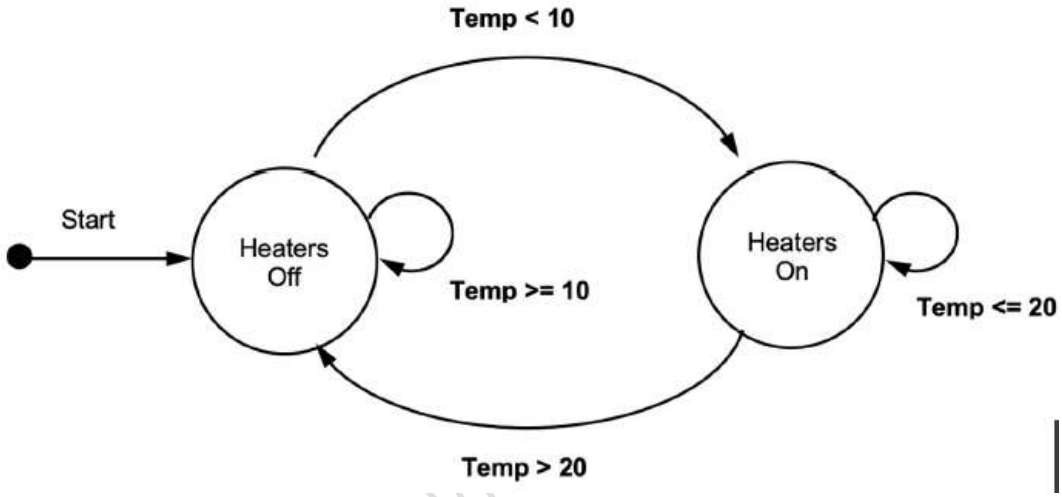
One mark per row (row 2 to 5):

Input	Output	Next state
		S1
Input-A	**Output-X**	**S2**
Input-A	**(none)**	S2
Input-B	Output-W	**S3**
Input-A	Output-W	**S4**

[4]

9618/21/O/N/21

4.



[3]

One mark for each label:

- Temp < 10 (Heaters Off to Heaters On)
- Temp > 20 (Heaters On to Heaters Off)
- on **BOTH** loops (non-contradictory values)

9618/22/M/J/24

5. (i)

Input	Output	Next state
		S1
Input-A	none	S3
Input-A	Output-W	**S3**
Input-B	none	**S2**
Input-B	**none**	**S5**
Input-A	**none**	**S2**
Input-A	**Output-X**	S4

One mark per row 3 to 7

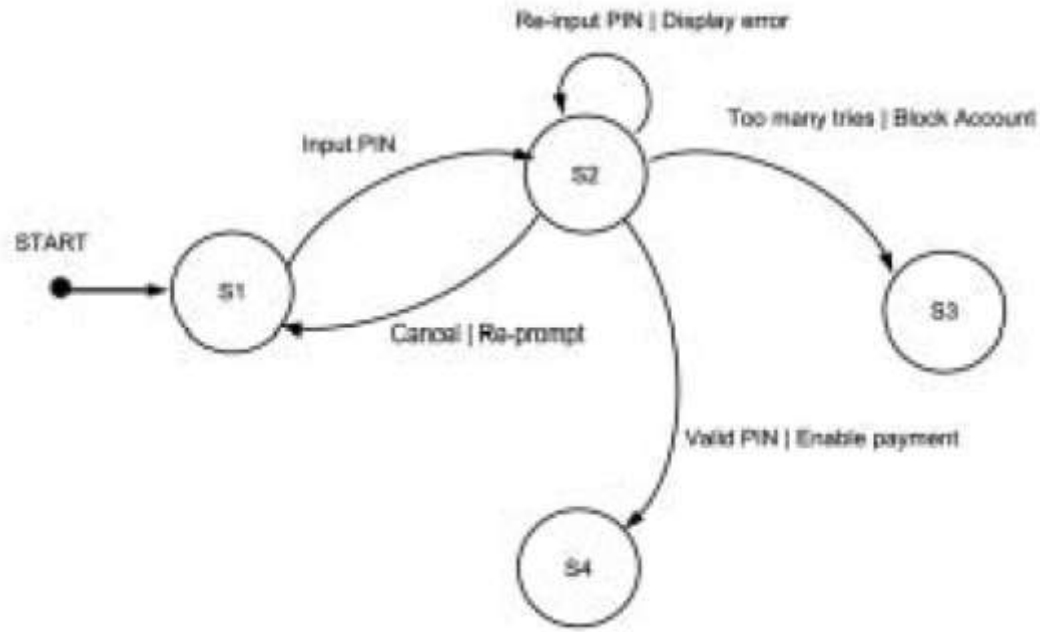
[5]

5 (ii)

Input-B, Input-A **[1]**

9618/22/M/J/23

6.



[4]

One mark for each:
1 Line from S1 to S2 **and** label
2 S2 loop label
3 S3 circle and label on incoming
4 S4 circle and label on incoming

7 (i)

The number of transitions that result in a different state	**3**
The number of transitions with associated outputs	**2**
The label that should replace 'X'	**Start**
The final or halting state	**S3**

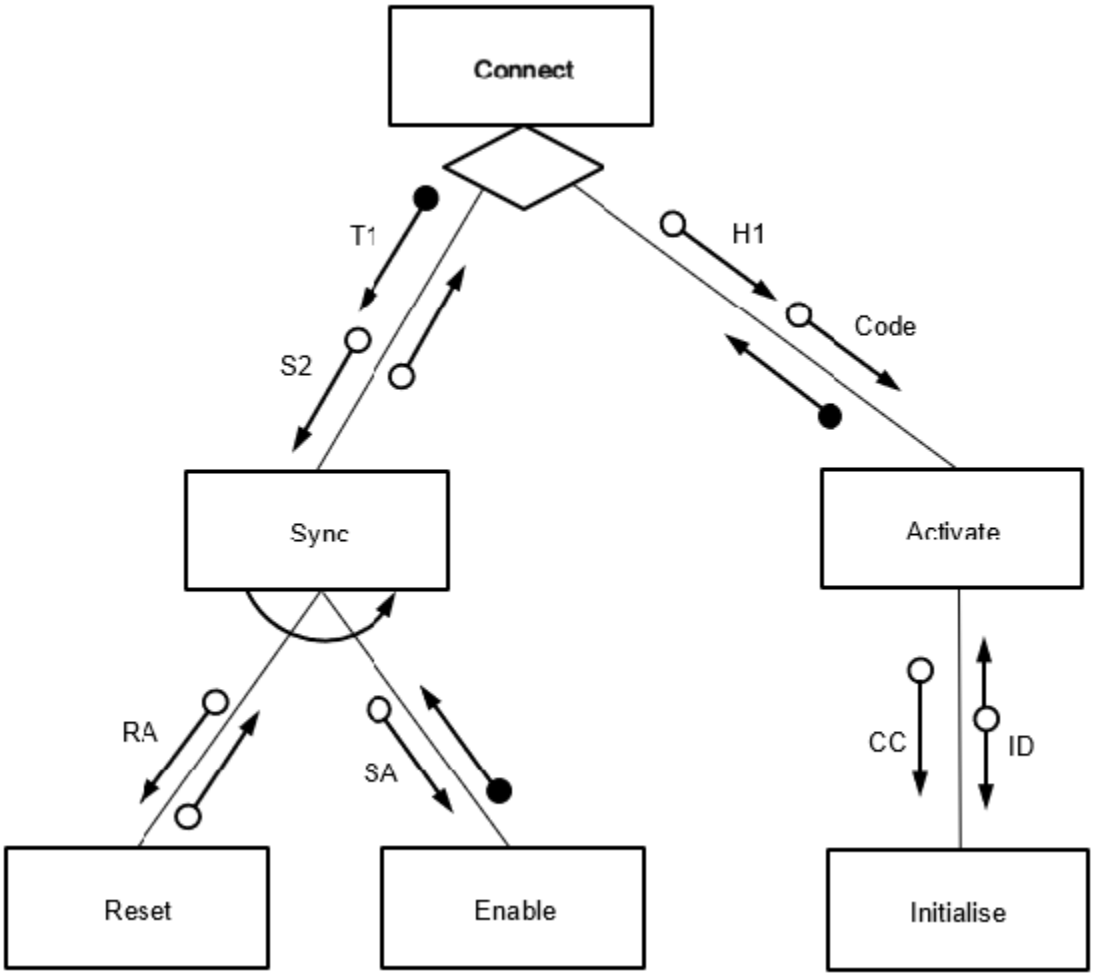
One mark per row **[4]**

7(ii)

Number of outputs: 1
Current state: S2

[2]

Answer
9618/21/O/N/24
8



[5]

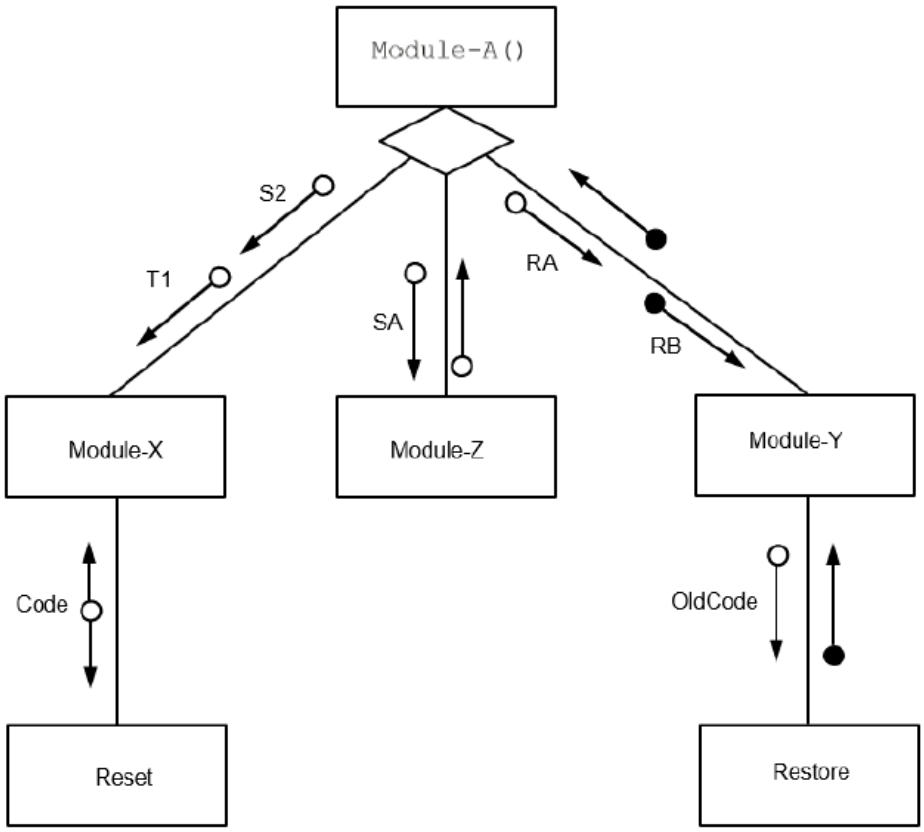
One mark per point:
1 All module identified in correct hierarchy
2 Parameters to both Sync and Activate **and** return values
3 Parameters to both Reset and Enable
4 Parameters to Initialise
5 Decision diamond

(b)
Explanation:
Means that Sync repeatedly calls Reset and (then) Enable
One mark for each point:
MP1: reference to iteration
MP2: naming all three modules correctly including correct sequence of calls

[2]

9618/23/O/N/23
9
MP1 iteration / looping
MP2 naming all four modules correctly in the correct sequence // e.g. `Module-A` repeatedly calls `Sub-Y1`, then `SubY2` then `Sub-9`

9618/22/O/N/23
10 (a)



[4]

One mark per point:
1 Module names
2 Parameters with labels to Module-X and between Module-X and Reset
3 Parameter (with label) to Module-Z and return from Module-Z
4 Parameters with labels to Module-Y and Restore and return values from Module-Y

(b)
Means that `Module-A` calls either one of `Module-X`, `Module-Y` or `Module-Z` (which one is called is decided at runtime).
One mark for reference to selection
One mark for naming all **four** modules correctly

[2]

9618/21/M/J/24

11 (i)

Means that Update calls (one of) either Sub-A, Sub-B or Sub-C

One mark for each point:

- reference to selection / decision / if
- naming all four modules correctly **[2]**

(ii)

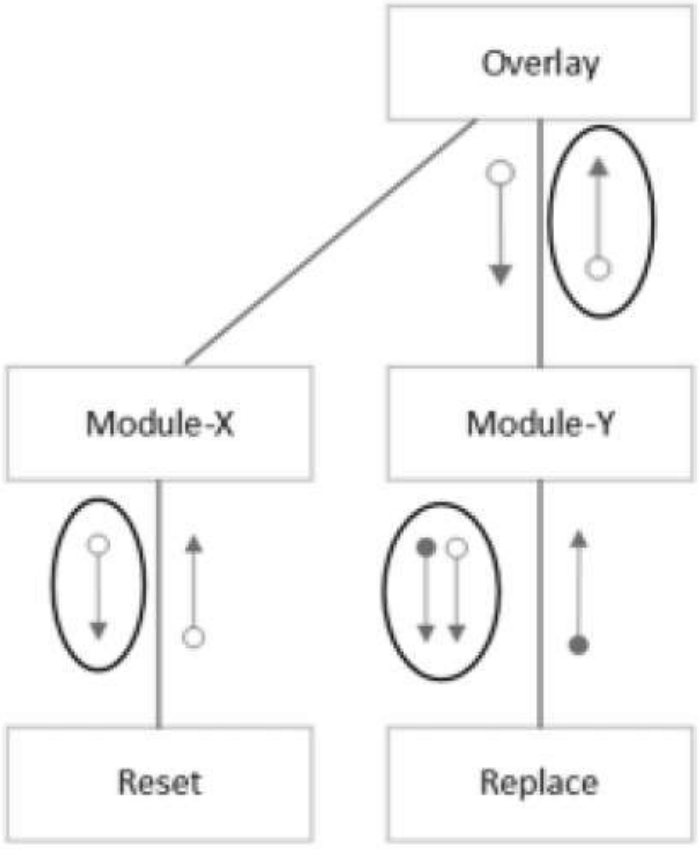
```
PROCEDURE Sub-A (Name : STRING, BYREF P2 : BOOLEAN)
FUNCTION Sub-B (P1 : REAL) RETURNS REAL
```

One mark per underlined part in each case **[4]**

9618/23/M/J/23

12.

Max 3 marks



[3]

MP1 All modules correctly labelled and interconnected.
MP2 Correct parameters and return type to Module-X and Reset
MP3 Correct parameters and return type to Module-Y and Replace
MP4 Correct parameters and return type to Overlay and Module-Y

9618/21/M/J/23

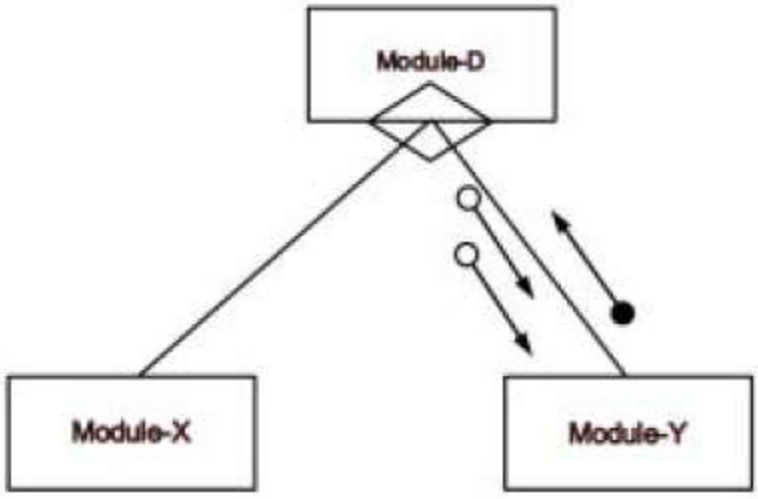
13(i)

One mark per point:

- Module-A calls the other three modules
- The process is repeated

9618/21/M/J/23

13. (ii)



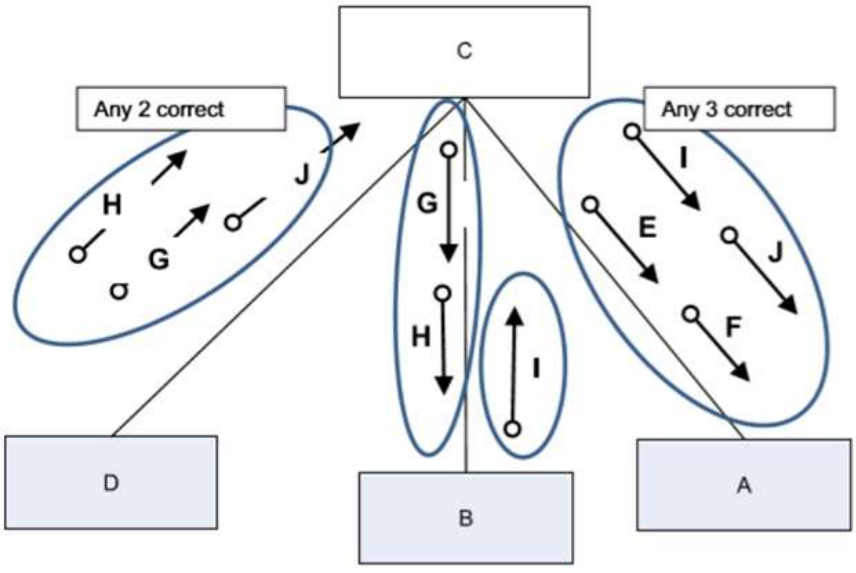
One mark per bullet point:

- All rectangles correctly labelled and interconnected
- All and only parameters as shown
- Selection diamond

[3]

9608 /MJ 2015/ 21

14.



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

[4] + [2]

Four marks for correct parameters
Two marks for correct modules.