**Q1.**

A particular Turing machine has states SB, S0, S1, SR and ST. SB is the start state and ST is the stop state. The machine stores data on a single tape which is infinitely long in one direction. The machine’s alphabet is 0, 1, #, x, y and □ where □ is the symbol used to indicate a blank cell on the tape.

The transition rules for this Turing machine can be expressed as a transition function δ. Rules are written in the form:

δ(Current State, Input Symbol) = (Next State, Output Symbol, Movement)

So, for example, the rule:

δ(SB, 1)  =  (S1, y, →)

means:

IF        the machine is currently in state SB AND the input symbol read from the tape is 1

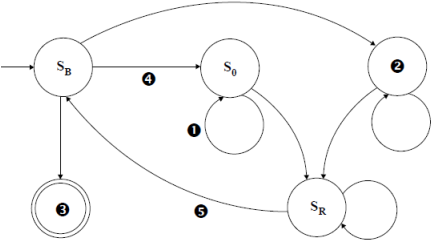
THEN  the machine should change to state S1, write a y to the tape and move the read/   
  write head one cell to the right

The machine’s transition function, δ, is defined by:

|  |  |
| --- | --- |
| δ (SB, 0)   =    (S0, x, →) δ (SB, 1)    =    (S1, y, →) δ (SB, #)    =    (ST, #, →)  δ (S0, 0)    =    (S0, 0, →) δ (S0, 1)    =    (S0, 1, →) δ (S0, #)    =    (S0, #, →) δ (S0, □)    =    (SR, 0, ←) | δ (S1, 0)    =    (S1, 0, →) δ (S1, 1)    =    (S1, 1, →) δ (S1, #)    =    (S1, #, →) δ (S1, □)    =    (SR, 1, ←)  δ (SR, 0)    =    (SR, 0, ←) δ (SR, 1)    =    (SR, 1, ←) δ (SR, #)    =     (SR, #, ←) δ (SR, x)    =    (SB, 0, →) δ (SR, y)    =    (SB, 1, →) |

**Figure 1** shows an unlabelled finite state transition diagram for this machine. Some of the state transition arrows represent more than one of the machine’s transition rules. For example, the arrow labeled ❶ represents the three rules: δ (S0, 0)  =  (S0, 0, →), δ (S0, 1)  =  (S0, 1, →) and δ (S0, #)  =  (S0, #, →).

**Figure 1**



(a)     (i)      Which states are represented by the labels ❷ and ❸ in **Figure 1**?

❷ S1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_                        ❸ ST\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(ii)     Which of the machine’s transition rule(s) is / are represented by the arrow labelled ❹ in **Figure 1**?

δ (SB, 0)   =    (S0, x, →)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(iii)    Which of the machine’s transition rule(s) is / are represented by the arrow labelled ❺ in **Figure 1**?

δ (SR, x)    =    (SB, 0, →)  
δ (SR, y)    =    (SB, 1, →)

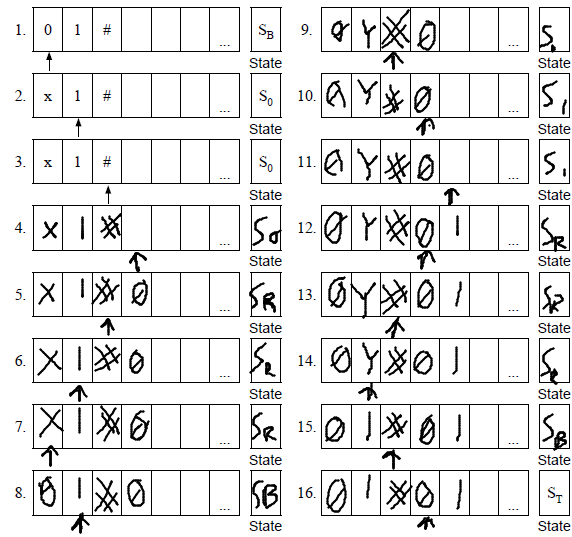
**(1)**

The machine's transition rule, δ, is repeated here so that you can answer part **(b)** without having to turn back in the question paper booklet.

|  |  |
| --- | --- |
| δ (SB, 0)    =    (S0, x, →) δ (SB, 1)    =    (S1, y, →) δ (SB, #)    =    (ST, #, →)  δ (S0, 0)    =    (S0, 0, →) δ (S0, 1)    =    (S0, 1, →) δ (S0, #)    =    (S0, #, →) δ (S0, □)    =    (SR, 0, ←) | δ (S1, 0)    =    (S1, 0, →) δ (S1, 1)    =    (S1, 1, →) δ (S1, #)    =    (S1, #, →) δ (S1, □)    =    (SR, 1, ←)  δ (SR, 0)    =    (SR, 0, ←) δ (SR, 1)    =    (SR, 1, ←) δ (SR, #)    =    (SR, #, ←) δ (SR, x)    =    (SB, 0, →) δ (SR, y)    =    (SB, 1, →) |

(b)     This Turing machine is carrying out a computation. The machine starts in state SB with the string 01# on the tape. All other cells contain the blank symbol, □ (not shown).

Trace the computation of the Turing machine, using the transition function δ. Show the contents of the tape, the current position of the read / write head and the current state as the input symbols are processed. The first three steps and final state have been completed for you.



**(6)**

(i)      Describe the purpose of the symbols x and y in this Turing machine’s alphabet.

Placeholders for the 0 or 1 being copied

**(1)**

(ii)     What does the Turing machine do?

This Turing machine copies everything before the hash symbol to after the hash symbol.

**(1)**

**(Total 11 marks)**

**Q2.**

A particular Turing machine has states S1, S2 and S3.  
S1 is the start state and S3 is the stop state.  
The machine uses one tape which is infinitely long in one direction to store data.  
The machine’s alphabet is 0, 1, o, e and □, where □ is the symbol used to indicate a blank cell on the tape.  
  
The transition rules for this Turing machine can be expressed as a transition function δ.   
Rules are written in the form:

δ(Current State, Input Symbol) = (Next State, Output Symbol, Movement)

So, for example, the rule:

δ (S1, 0) = (S1, 0, →)

means

|  |  |
| --- | --- |
| IF | the machine is currently in state S1 AND the input symbol read from the tape is 0 |
| THEN | the machine should remain in state S1, write a 0 to the tape and move the read/write head one cell to the right |

The machine’s transition function, δ, is defined by:

δ (S1, 0) = (S1, 0,→)

δ (S1, 1) = (S2, 1, →)

δ (S1, □) = (S3, e, →)

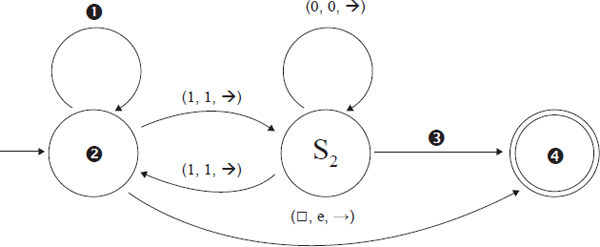
δ (S2, 0) = (S2, 0, →)

δ (S2, 1) = (S1, 1, →)

δ (S2, □) = (S3, 0, →)

The diagram below shows a partially labelled finite state transition diagram for this machine.

Some labels are missing and have been replaced by numbers such as . Each state transition arrow is labelled with the input symbol, the output symbol and the direction of movement, in that order. For example (□, e, →) means that if the input symbol is □, an e is written to the tape and the read/write head moves right one cell.



(a)     Four labels are missing from the diagram above.

Write the missing labels in the table below.

|  |  |
| --- | --- |
| **Number** | **Correct Label** |
|  |  |
|  |  |
|  |  |
|  |  |

**(2)**

(b)     The Turing machine is carrying out a computation using one tape which is infinitely long in one direction. The machine starts in state S1 with the string 01100 on the tape.   
All other cells contain the blank symbol, □. The read/write head is positioned at the leftmost zero, as indicated by the arrow.

|  |  |  |
| --- | --- | --- |
|  | Current State: |  |

Trace the computation of the Turing machine, using the transition function δ. Show the contents of the tape, the current position of the read/write head and the current state as the input symbols are processed.

|  |  |  |
| --- | --- | --- |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |

**(4)**

(c)     What is the purpose of the algorithm represented by this Turing machine?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(d)     Explain the importance of the theory of Turing machines to the subject of computation.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(2)**

**(Total 9 marks)**

**Q3.**

A particular Turing machine has states S1, S2, S3 and S4. S1 is the start state and S4 is the stop state. The machine uses one tape which is infinitely long in one direction to store data. The machine’s alphabet is 1, □. The symbol □ is used to indicate a blank cell on the tape.

The transition rules for this Turing machine can be expressed as a transition function δ. Rules are written in the form:

δ(Current State, Input Symbol) = (Next State, Output Symbol, Movement)

So, for example, the rule:

δ (S1, 1) = (S1 , 1, → )

means:

|  |  |
| --- | --- |
| IF | the machine is currently in state S1 AND the input symbol read from the tape is 1 |
| THEN | the machine should remain in state S1, write a 1 to the tape and move the read/write head one cell to the right |

The machine’s transition function, δ, is defined by:

δ (S1, 1) = (S1 , 1, → )

δ (S1, □) = (S2 , □, ← )

δ (S2, 1) = (S3 , □, ← )

δ (S3, 1) = (S4 , □, ← )

(a)     The Turing machine is carrying out a computation. The machine starts in state S1 with the string 1111 on the tape. All other cells contain the blank symbol, □. The read/write head is positioned at the leftmost 1, as indicated by the arrow.

|  |  |  |
| --- | --- | --- |
|  | Current State: |  |

Trace the computation of the Turing machine, using the transition function δ. Show the contents of the tape, the current position of the read / write head and the current state as the input symbols are processed.

|  |  |  |
| --- | --- | --- |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |
|  | Current State: |  |

**(6)**

(b)     Explain what this Turing machine does.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(c)     Explain what a *Universal Turing machine* is.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(2)**

**(Total 9 marks)**

**Q4.**

A Turing machine has been designed to recognise palindromic binary numbers, ie numbers such as 101 and 0110 that read the same from left to right as from right to left.

The machine has states SB, S0, S1, SC0, SC1, SL, SY and SN.  
SB is the start state and SY and SN are the stop states.

The machine stores data on a single tape which is infinitely long in one direction. The machine’s alphabet is 0, 1 and **□**, where **□** is the symbol used to indicate a blank cell on the tape. The machine will enter state SY if the value represented on the tape is a palindromic binary number, otherwise it will enter state SN.

The transition rules for this Turing machine can be expressed as a transition function δ.  
Rules are written in the form:

δ(Current State, Input Symbol) = (Next State, Output Symbol, Movement)

So, for example, the rule:

δ (SB, 0) = (S0 , **□**, →)

means:

|  |  |
| --- | --- |
| IF | the machine is currently in state SB AND the input symbol read from the tape is 0 |
| THEN | the machine should change to state S0, write a blank symbol (**□**) to the tape and move the read / write head one cell to the right |

The machine’s transition function, δ, is defined by:

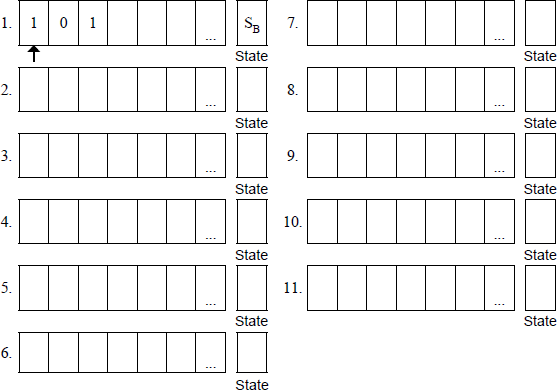
|  |  |  |
| --- | --- | --- |
| δ (SB, 0) = (S0, □, →)      δ (SB, 1) = (S1, □, →)      δ (SB, □) = (SY, □, →) |  | δ (SC0, 0) = (SL, □,←) δ (SC0, 1) = (SN, 1,←) δ (SC0, □) = (SY, □,→) |
| δ (S0, 0) = (S0, 0, →) δ (S0, 1) = (S0, 1, →)  δ (S0, □) = (SC0, □, ←) |  | δ (SC1, 0) = (SN, 0, ←) δ (SC1, 1) = (SL, □, ←) δ (SC1, □) = (SY, □, →) |
| δ (S1, 0) = (S1, 0, →) δ (S1, 1) = (S1, 1, →)  δ (S1, □) = (SC1, □, ←) |  | δ (SL, 0) = (SL, 0, ←) δ (SL, 1) = (SL, 1,←) δ (SL, □) = (SB, □,→) |

(a)     This Turing machine is carrying out a computation. The machine starts in state SB with the string 101 on the tape. All other cells contain the blank symbol, **□** (not shown).  
The read / write head is located at the left hand symbol of the string and is indicated with an upward arrow.

Trace the computation of the Turing machine, using the transition function δ.

Show the contents of the tape, the current position of the read/write head and the current state as the input symbols are processed.

The initial configuration of the machine has been completed for you in step 1.



**(5)**

(b)     The three rules shown below are part of the machine’s transition function.

Explain what effect these three rules, taken together, have on the tape, the read / write head and the state of the Turing machine:

δ (S0, 0)   =    (S0, 0, →)  
δ (S0, 1)   =    (S0, 1, →)  
δ (S0, **□**)   =    (SC0, □,←)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(2)**

(c)     A Universal Turing machine (UTM) is a special type of Turing machine that can be considered to act like an interpreter.

Explain how a UTM can be considered to be an interpreter.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

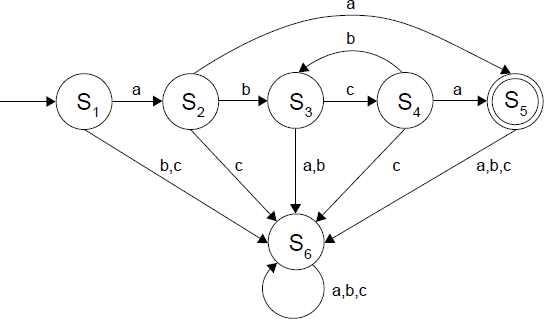
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(2)**

**(Total 9 marks)**

**Q5.**

The diagram below shows a Finite State Automaton (FSA). The FSA has input alphabet {a, b, c} and six states, S₁, S₂, S₃, S₄, S₅ and S₆.



(a)     Complete the empty cells in the **part of the transition table shown** below for the FSA in the diagram.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Current State** | S₁ | S₁ | S₁ | S₂ | S₂ | S₂ | S₃ | S₃ | S₃ |
| **Input Symbol** | a | b | c | a | b | c |  |  |  |
| **Next State** | S₂ | S₆ | S₆ | S₅ | S₃ | S₆ |  |  |  |

**(1)**

(b)     Give the name of the state that the FSA will end up in when processing the string abcb.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(c)     Describe the purpose of state S₆.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(d)     The FSA in the diagram above accepts strings that consist of:

•        a letter a

•        followed by zero or more occurrences of the string bc and

•        ending with a second letter a.

For any FSA, it is possible to write a regular expression that will match the same language (set of strings) as the FSA.

Write a regular expression that will match the same language that is accepted by the FSA above.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(2)**

(e)     The Turing Machine is a more powerful abstract model of computation than the FSA.

Explain why the Turing Machine model can be used to recognise a greater range of languages than an FSA could.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

**(Total 6 marks)**