Turing Machines

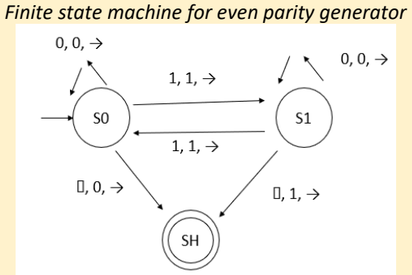
Purpose of Turing Machines

- Turing machines are a model of computation that help us understand how algorithms can be solved computationally.
- If a problem is computable then it can be solved by a Turing machine (Church-Turing thesis).
- Turing machines can be used determine whether an algorithm is computable.

How a Turing Machine Works

- A Turing machine is a finite state machine with a tape of infinite length that is divided into squares. This is the memory of the machine.
- Has a finite set of symbols, commonly 0 , 1 and □ which indicates no value. Each square on the tape takes on one of the values of the symbols.
- Has a head which can read and write to the tape and move along the tape in either direction.
- Has a finite set of states. It can have a start state and must have a halting state.
- Behave as interpreters because the deal with one instruction at a time.
- Turing machines can be expressed using:
  - Finite State Machines / diagrams
  - State transition tables
  - State transition functions

Finite state machine for even parity generator



First value is the symbol read, Second value is the symbol to write and third value is the direction in which to move the head

State transition table for even parity generator

State	Read	Write	Move	Next state
S0	0	0	→	S0
S0	1	1	→	S1
S0	□	0	→	SH
S1	0	0	→	S1
S1	1	1	→	S0
S1	□	1	→	SH

State transition function for even parity generator

ð(current state, input symbol) = (next state , output symbol ,direction)

ð(S0,0) = (S0,0,→)
ð(S0,1) = (S1,1,→)
ð(S0,□) = (SH,0,→)
ð(S1,0) = (S1,0,→)
ð(S1,1) = (S0,1,→)
ð(S1,□) = (SH,1,→)

Worked example:

	The green arrow denotes the position of the read/ write head
Step 1	□ 1 0 1 1 1 0 □ □ □ S0
Step 2	□ 1 0 1 1 1 0 □ □ □ *S1*
Step 3	□ 1 0 1 1 1 0 □ □ □ *S1*
Step 4	□ 1 0 1 1 1 0 □ □ □ *S0*
Step 5	□ 1 0 1 1 1 0 □ □ □ *S1*
Step 6	□ 1 0 1 1 1 0 □ □ □ *S0*
Step 7	□ 1 0 1 1 1 0 □ □ □ *S0*
Step 8	□ 1 0 1 1 1 0 0 □ □ *SH*

Universal Turing Machines

- For each operation a different Turing machine has to be created, so this is not ideal.
- For a Turing machine, the state transition diagram / function / FSM are the instructions so is separate from the tape
- A Universal Turing machine is a Turing machine that can execute another Turing machine. The instructions of the Turing machine are stored on the tape.
- This model of computing is what is used in modern computers today where both the program instructions and the data are stored in memory.

Tape used for even parity generator	□ 1 0 1 1 1 0 □ □ □