

1. [4 pts] Exam the formal definition of the Turing-Machine to answer the following question:

1.1) Can a TM ever write a blank symbol $_$ on its type?

- **Yes, a Turing machine can write a blank symbol on its tape. In a turing machine, the blank symbol is in the set of the tape alphabet which means you would be allowed to write it into the tape.**

1.2) Can the tape alphabet Γ be the same as input alphabet Σ ?

- **No, the tape alphabet can not be the same as the input alphabet because the blank symbol is in the set of the tape alphabet but never in the set of the input alphabet. This one character will make it such that the tape alphabet and input alphabet can never be the same because only the tape alphabet can ever have access to the blank symbol.**

1.3) Can the Turing machine's head ever be in the same location in two successive steps?

- **Yes, the Turing machine's head can be in the same location for 2 successive steps. A blank symbol can not be placed to the left of the head when the head is in the leftmost possible tape position because you can not move left from the leftmost position in a turing machine leading to there being successive steps in which the head is in the same position.**

1.4) Can a Turing machine just contain a single state?

- **No, a Turing machine can not just contain a single state because the accepting state and the rejecting state must be different which means that there must be greater than or equal to 2 states.**

2. [6 pts] Give the informal description of the TM that decides the following language, assume $\Sigma = \{0, 1\}$. (Note: check textbook pp. 160, TM M1 for an example of such description)

$L = \{w \mid w \text{ contains equal number of 0s and 1s}\}$

1. **Start with an initial state**
2. **Go from the head and if the first character is a 0 do steps 3 - 5**
3. **Replace the 0 with an X and move R till you see the first 1**
4. **Once you see the first 1, you replace it with a Y and then move L**
5. **Move left until you see an X, then move right till you see another 0, replace it with an X, and continue the above rules**
6. **If the first character is a 1, replace it with Y and do the same steps of 2 - 5 but flip the 0s/Xs and 1/Ys. For example, if you start with a 1, replace it with a Y and move R, and the first 0 you make an X and move L. Repeat this process until you reach the end.**
7. **Once you hit the blank string, go L and see if there are any 1s or 0s in which case it goes to the rejecting state**
8. **If you go L and there are no 0s and 1s then there is an equal amount of each so you would go to the accepting state**
9. **END**