

# Languages and Automata Assignment 4

Dibran Dokter s1047390

March 6, 2020

## 1

a)

$\mathcal{L}_1$  is not regular since it is similar to the language  $\{a^i b^i \in A^* | i \in N\}$ , in that it has words of the same length from  $A^*$ . However we have already seen that this is a language that is not regular.

b)

$\mathcal{L}_2$  is not regular since it is impossible to ensure that  $a^p$  and  $b^p$  have an equal length. In this way it is similar to the language  $\{a^i b^i | i \in N\}$  which is proven to not be regular.

c)

$\mathcal{L}_3$  is not regular because to check all the possible combinations we would need an infinite number of states since we would need to check every combination where  $n \neq m$ , this can be seen since it is similar to the language  $\{a^j b^i | j, i \in N\}$  which is not regular.

d)

since L is regular we know that the complement of L is also regular. however we do not know whether K is regular. But since we describe only the words that are both in L and K because of the intersection the statement holds.

## 2

a)

when  $m$  goes to infinity, then  $n$  does as well, which means that we need a massive number of states to check for all distinguishable options, leading to an infinite number of states when  $m$  goes to infinity. This means there cannot be a DFA that describes it since it would need to be infinite and thus the language is not regular.

b)

To create a DFA for this language we would need to include all the possible options for the subwords for every possible word of every length. This again leads to an infinity number of distinguishable words and thus there cannot exist a DFA for this language and it is not regular.

### 3

First we make sure that every state is reachable, which is the case since we can reach every state from  $q_0$ . After this we identify the states that are language equivalent, this leads to the following states:  $\{q_0, q_1, \{q_2, q_3\}, q_4\}$ . After we group these language equivalent states together we get the following minimal DFA:

