

Theory of Computation 2019

$(A \leftrightarrow B)$

$$1a) i) (B \rightarrow (A \rightarrow C)) \leftrightarrow (A \rightarrow (B \rightarrow C)) \quad A \rightarrow B \wedge B \rightarrow A$$

$$(\neg B \vee (\neg A \vee C)) \leftrightarrow (\neg A \vee (\neg B \vee C))$$

$$(\neg B \vee (\neg A \vee C)) \rightarrow (\neg A \vee \neg B \vee C) \wedge (\neg A \vee (\neg B \vee C)) \rightarrow (\neg B \vee \neg A \vee C)$$

$$(\neg (\neg B \vee \neg A \vee C) \vee (\neg A \vee \neg B \vee C)) \wedge \neg (\neg A \vee \neg B \vee C) \vee (\neg B \vee \neg A \vee C)$$

$$((B \vee A \vee \neg C) \vee (\neg A \vee \neg B \vee C)) \wedge ((A \vee B \vee \neg C) \vee (\neg B \vee \neg A \vee C))$$

| \wedge |

The propositional formula is satisfiable and valid. When trying to construct DNF form, the formula reduces to true, hence it is satisfiable and valid.

is satisfiable and valid

$$\begin{aligned} \text{ii}) \quad & -((\bar{a} \cdot \bar{b}) + -(\bar{c} \cdot \bar{a})) & -((\bar{a} \cdot \bar{b}) + (\bar{c} + \bar{a})) \\ & = -(\overline{(\bar{a} + b)} + (\bar{c} + a)) & -((\bar{a} \cdot \bar{b}) + \overline{(\bar{c} \cdot \bar{a})}) \\ & = (\bar{a} + b) \cdot \overline{(\bar{c} \cdot \bar{a})} & -((\bar{a} + \overline{(\bar{c} \cdot \bar{a})}) \cdot (\bar{b} + \overline{(\bar{c} \cdot \bar{a})})) \\ & = (\bar{a} + b) \cdot \overline{(\bar{c} + a)} & -((\bar{a} + \bar{c} + a) \cdot (\bar{b} + \bar{c} + a)) \\ & = (\bar{a} \cdot \overline{(\bar{c} + a)}) + (b \cdot \overline{(\bar{c} + a)}) & \overline{(\bar{a} + \bar{c} + a)} + \overline{(\bar{b} + \bar{c} + a)} \\ & = (\bar{a} \cdot (\bar{c} \cdot \bar{a})) + (b \cdot (\bar{c} \cdot \bar{a})) & (\bar{a} \cdot \bar{c} \cdot \bar{a}) + (b + \bar{c} + \bar{a}) \\ & = b \cdot c \cdot \bar{a} \end{aligned}$$

$$\text{iii) } f(a \rightarrow b) = \bar{a} + b$$

$$f(\neg a) = \overline{f(a)}$$

$$f(a \vee b) = f(a) + f(b)$$

$$f(a \wedge b) = f(a) \cdot f(b)$$

$$f(a \leftarrow \rightarrow b) = f(a \rightarrow b) \cdot f(b \rightarrow a)$$

$$f(p) = p$$

b) i) Nobody is friends with Mark

$$\forall x (\neg(x = \text{Mark}) \wedge \neg F(x, \text{Mark}))$$

ii) Olivia isn't friends with anybody who isn't friends with her

$$\forall x (\neg(x = \text{Olivia}) \wedge \neg F(\text{Olivia}, x) \rightarrow \neg F(x, \text{Olivia}))$$

iii) Mark is friends with all of his friends' friend

$$\forall x \forall y (\neg(x = \text{Mark}) \vee y = \text{Mark}) \wedge \neg(x = y) \wedge (F(\text{Mark}, x) \wedge F(x, y))$$

Olivia has at least three friends

$$\neg \exists x \forall y \neg F(\text{Mark}, y)$$

$$\begin{aligned} \text{iv)} \quad & \exists x \exists y \exists z (\neg(x = \text{Olivia} \vee y = \text{Olivia} \vee z = \text{Olivia}) \wedge \neg(x = y \vee y = z \vee x = z) \\ & \wedge F(\text{Olivia}, x) \wedge F(\text{Olivia}, y) \wedge F(\text{Olivia}, z)) \end{aligned}$$

$$\text{v)} \exists x \forall y \exists z (\neg(x = y) \wedge \neg(y = z) \wedge \neg(x = z) \wedge F(x, y) \wedge F(y, z))$$

$$c) M \models \forall x \forall y. x < y \rightarrow \exists z. x < z \wedge z < y$$

Since the domain is \mathbb{R}

Proof by contradiction,

(This might be super wrong)

Assume that $\forall x \forall y. x < y$ is false

that means that there are no values $\in \mathbb{R}$

that are between two values than one another

then $\exists z. x < z \wedge z < y$ would always be false

Since from our assumption all the values are not less or more than one another.

but we know that $\exists n (x < n < y)$ whenever $n \in \mathbb{R}$

hence $M \models \forall x \forall y. x < y \rightarrow \exists z. x < z \wedge z < y$

d)

a) Base Case: $\text{ins } x [] = x :: []$

$$\forall x \text{ ins } x xs = x$$

IH: $\text{ins } x (y :: ys) = x :: (y :: ys)$

Step Case: $\text{ins } z (\text{ins } x (y :: ys))$

$$= \text{ins } z (\text{ins } x (y :: ys)) \text{ IH}$$

$$= z :: x :: y :: ys$$

each number is appended to the list

b) Base Case: $\text{ins } x \ [] = x :: []$

IH: $\text{ins } x \ (y :: ys) = \begin{cases} x :: (y :: ys) & \text{if } x \leq y \\ y :: (\text{ins } x (ys)) & \text{if } x > y \end{cases}$

Step Case: $\text{ins } z \ (\text{ins } x (y :: ys))$

let $z = x + 1$

$\text{ins } z - 1 (x :: y :: ys)$

Since $x - 1 \leq x$

$= x - 1 :: x :: y :: ys$

let $z = x + 1$ and $x + 1 \leq$

$\text{ins } z \ (x :: y :: ys)$

$= x :: (\text{ins } x + 1 (y :: ys))$

$$= x :: (\text{ins } x+1 \ (y :: ys))$$

e) $M = (V, R) \quad \exists y. Rxy \wedge Ryx$

Suppose $M = (V, R)$ is not dense; we have to show

$M, w \models_p \Box \Box A \rightarrow \Box A$ is not valid. Assume $M, w \models_p \Box \Box A$ is true

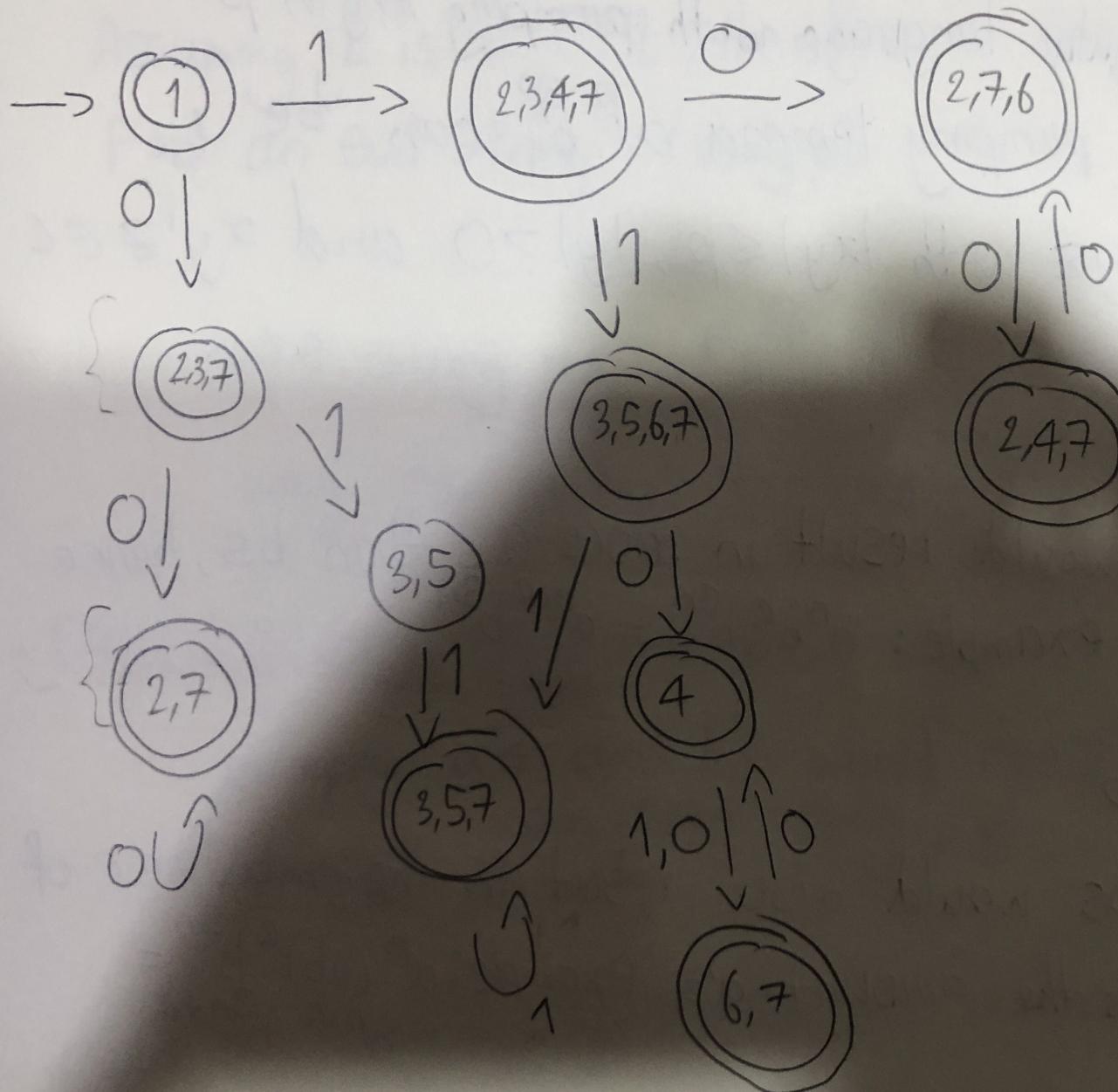
Since M is not dense we have Rww' and $Rw'w''$
but not Rww''

$M, w \models_p \Box \Box A$ holds, but $M, w \models_p \Box A$ does not
as there are no Rww'' but A is only valid at

w''
thus $M, w \models_p \Box \Box A$ and $M, w' \models \Box A$ holds, but
 $M, w \not\models_p \Box A$

Thus the frame has to be dense for $\Box \Box A \rightarrow \Box A$ to
be valid

2. a)



b) i)

Σ yes

Output derivation

B $S \rightarrow B$

Σ $B \rightarrow \Sigma$

i) bac

not generatable

abcc

not generatable

abc

$S \rightarrow ABC$

ab

$S \rightarrow B$

abc

$B \rightarrow bBc$

ii) $\{a^i b^j c^{i+j} \mid i \geq 0, j \geq 0\}$

Empty or in alphabetical order where the number of a's and b's equal to c

c) $L = \{a^n b^{3n} \mid n \geq 0\}$

Assume L is a regular language with pumping length p .
 $a^p b^{3p} \in L$. By pumping lemma $a^p b^{3p}$ can be decomposed into xyz with $|xy| \leq p$, $|y| > 0$ and $xy^i z \in L$ when $i \geq 0$.

Case 1: $y = a$

pumping a 's would result in more a 's than b 's, hence not in L example: $a^n a^k b^{3n} = a^{\frac{n+k}{n} k} b^{3n}$

Case 2: $y = ab$

pumping ab 's would also result in the imbalance of b 's being 3 times the amount a 's example: $a^n (ab)^k b^{3n} = a^{\frac{n+k}{n} k} b^{3n+k}$

Case 3: $y = b$

pumping b 's would result in b 's being more than 3 times the number of a 's example: $a^n b^k b^{3n} = a^n b^{3n+k}$

In all cases it can't satisfy the pumping lemma, thus

$a^n b^{3n}$ is not regular

$$d) L = \{a^n b^{3n} | n \geq 0\}$$

Assuming L is a CFL, therefore it has a pumping length p .
Find an "evil string" (using p)

$w \in L$

take string as $a^p b^{3p}$

vxy can contain 3 cases

Case 1: $vxy = a's$ and $b's$

pumping $a's$ and $b's$ would cause an imbalance

example: $a^{p+n} b^{3p+n} \frac{3p+n}{p+n} \neq 3$

case 2: $vxy =$ only 1 letter $a's$ or $b's$

pumping only 1 letter would cause an imbalance

for example $a^{p+n} b^{3p}$ or $a^p b^{3p+n}$

$$\frac{3p}{p+n} \neq 3 \quad \text{or} \quad \frac{3p+n}{p} \neq 3$$

Hence in all the cases it cannot satisfy the pumping lemma

thus $a^n b^{3n}$ is not contextfree

$$e) \bar{L} = \{w \mid w \notin L\}$$

The turing machine cannot be created because
a turing machine acceptance is undecidable.

Proof by contradiction

Assuming the input turing machine is decidable

let H be a decider

$$H(\langle M, w \rangle) = \begin{cases} \text{accept} & M \text{ accepts } w \\ \text{reject} & M \text{ rejects } w \end{cases}$$

now we have another TM D that will negate the
output of turing machine M

essentially $D(M) = \neg H(M, \langle M \rangle)$

If we run D on D then

$$D(\langle D \rangle) = \begin{cases} \text{accept} & D \text{ does not accept } \langle D \rangle \\ \text{reject} & D \text{ does accept } \langle D \rangle \end{cases}$$

Hence D accepts and does not accept D, which
leads to a contradiction.

Thus a TM that decides a language L is not
a decidable language.