1A (1) If ␣ is in current cell, then REJECT

(2) If a is in current cell, then erase and move right, goto (5)

(3) If 1 is in current cell, then erase and move right, goto (9)

(4) If p is in current cell, then erase and move right goto (1).

(5) If 1 is in current cell, then ACCEPT.

(6) If ␣ is in current cell, then REJECT.

(7) If p is in current cell, the erase and go right.

(8) If a is in current cell, then erase and go right, goto (5)

(9) If a is in current cell, the ACCEPT.

(10) If ␣ is in current cell, the REJECT.

(11) If p is in current cell, then erase and go right.

(12) If p is in current cell, then erase and go right.

1 a) P → $\sqcup$ → REJECT

al → $\sqcup$l → ACCEPT

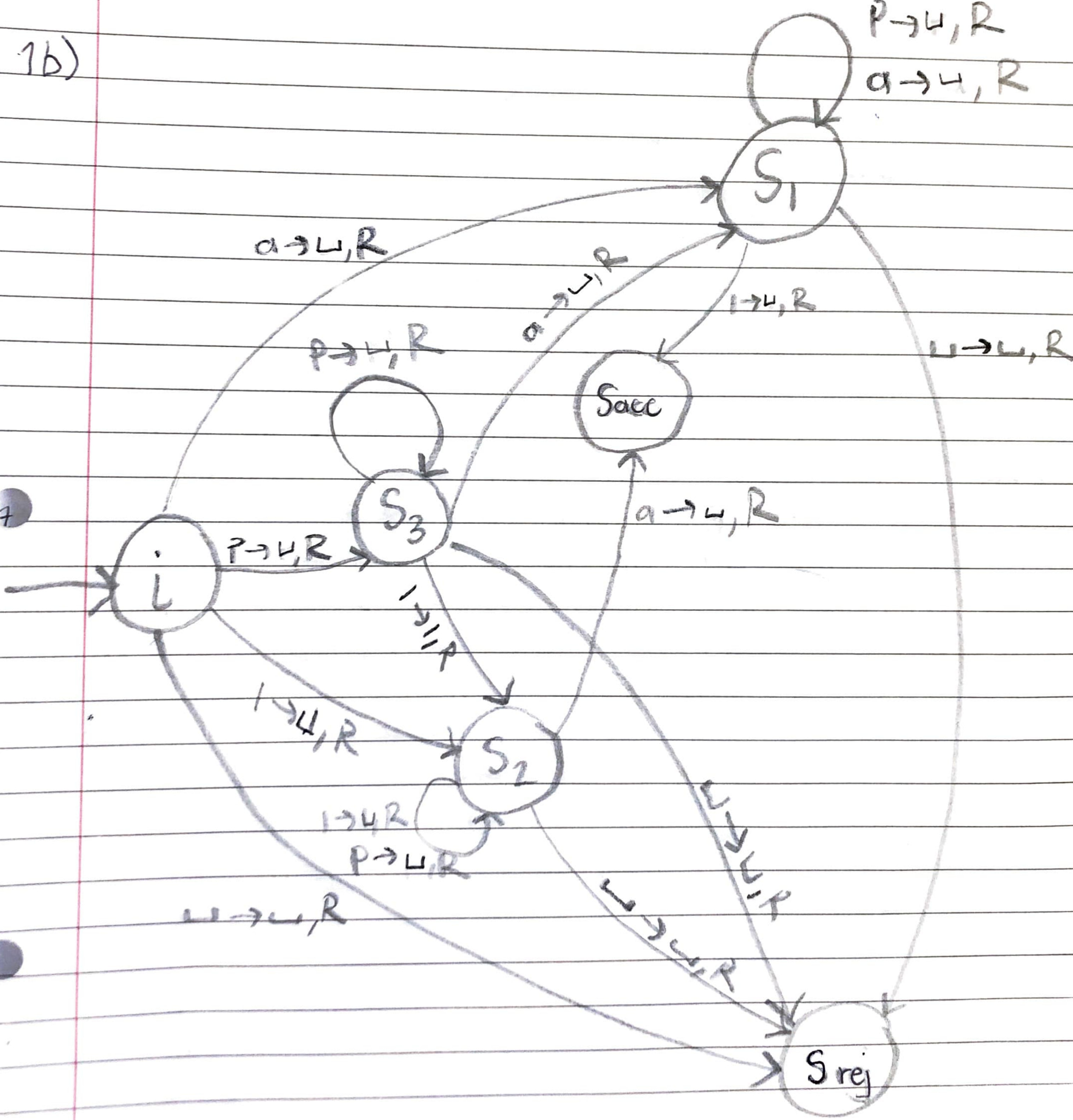
pap → $\sqcup$ap → $\sqcup\sqcup$p → $\sqcup\sqcup\sqcup$ → REJECT

pla → $\sqcup$la → $\sqcup\sqcup$a → ACCEPT

aapppla → $\sqcup$apppla → $\sqcup\sqcup$pppla → $\sqcup\sqcup\sqcup$ppla

→ $\sqcup\sqcup\sqcup\sqcup$pla → $\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup$la → ACCEPT.

1b)



2 $A^* = \{0, 1, 2, 3, 4 \dots\}$
$\quad\quad\;\; \| \;\; \| \;\; \| \;\; \| \;\; \|$
$\quad\quad\; w_1 \; w_2 \; w_3 \; w_4 \; w_5$

$m=0 \quad 3(0)+1 = 1$ (not even) ×

$m=1 \quad 3(1)+1 = 4$ (even) ✓

$m=2 \quad 3(2)+1 = 7$ (not even) ×

$m=3 \quad 3(3)+1 = 10$ (even) ✓

.....

$m=(5) \quad 3(5)+1 = 16$ ✓

$$\underbrace{\overset{m=1}{4} \quad \overset{m=3}{10}}_{6} \quad \underbrace{\overset{m=5}{16}}_{6} \quad \underbrace{\overset{m=7}{22}}_{6} \quad \underbrace{\overset{m=9}{28}}_{6}$$

$w_i = i-1$ i.e. $w_2 = 2-1 = 1$

want $i-1 = 6m + 4$
want $i = 6m + 5$

multiplying by an even number always gives an even number and adding one to an even number always gives an odd number. odd numbers $\notin L$. Only when m is an odd number then it's $\in L$.

2. E = Given the input tape $A^* = \{w_1, w_2 \dots\}$

1. Repeat the following for $i = 1, 2, 3 \dots$

2. At step i, if $i = 6m+5$ for some $m \in \mathbb{N}$, then print out w_i.

3 a) $E_{PSG} = \{ \langle G \rangle \mid G$ is a phrase structure grammar and $L(G) = \emptyset \}$

We define the turing machine as
M = on input ⟨G⟩, where G is a CFG:

1. Mark all terminal symbols in G

2. Repeat until no new variable get marked.

3. A production rule for a phrase structure grammar on the left can have any string of terminals and nonterminals provided ∃ at least one nonterminal. Mark any nonterminal on the left if ~~G~~ ~~context~~ each symbol (terminal or nonterminal) on the right has already been marked.

4. If the start symbol ⟨S⟩ is not marked, then accept; otherwise, reject.

3(b) The language

$E_{PSG} = \{\langle G \rangle \mid G$ is a phase structure grammer and $L(G) = \emptyset\}$

consists of two conditions,

→ G is a PSG

→ $L(G) \neq \emptyset$

Let C be the set of all Turing recognizable languages over a given finite alphabet A. As proven, the set of all Turing machines is countably infinite, so C must also be countably infinite. Clearly $E_{PSG} \subsetneq C$ as the condition $L(B) = \emptyset$ shrinks the size of the set. E_{PSG} is thus a subset of a countably infinite set, so it could be finite or countably infinite. In tutorial 22, we prove that the language $E_{DFA} = \{\langle B \rangle \mid B$ is a DFA and $L(B) = \emptyset\}$ is countably infinite. $E_{DFA} \subseteq E_{PSG}$. We have shown E_{PSG} has a countably infinite subset. We conclude E_{DFA} must be countably infinite.