The class of TM-decidable languages is closed under...

Union:

Let L_1 and L_2 be decidable languages
TMs M_1 and M_2 decide them
let TM M decide the language $L: L_1 \cup L_2$

On input ω:
① Run M_1 on ω → if accept then ACCEPT
② Run M_2 on ω → if accept then ACCEPT
otherwise → REJECT

TM M ACCEPTS ω if
→ M_1 or M_2 ACCEPT
TM M REJECTS ω if
→ M_1 and M_2 both REJECT
M decides $L_1 \cup L_2$

if $\omega \in L_1$ or L_2
M will Accept ω
$L_1 \cup L_2$ is Decidable

Concatenation:

Let L_1 and L_2 be decidable language
concatenation → $L_1L_2 = \{xy \mid x \in L_1 \text{ and } y \in L_2\}$
let TMs M_1 and M_2 decide L_1 and L_2
let TM M decide L_1L_2

On input ω:
① partition ω into 2 substrings xy
② Run M_1 on x
③ Run M_2 on y
if both accept → M ACCEPTs ω

(Partition must be adjusted + rerun through M, if none accept after trying all → REJECT)

TM → accepts when both M_1 and M_2 accept.

L_1L_2 is decidable

Star:

for language L_1
$L^* = \{x \in L \cup LL \cup LLL \cup \ldots\}$ let M_1 decide L
let M decide L^*

On input ω:
① For each possible partition of ω, $\omega_1\omega_2\omega_3\omega_4 \cdots \omega_n$
② Run M_1 on $\omega_1 \ldots \omega_n$ (each subsection)
③ if M_1 accepts each ω_i → ACCEPT
otherwise → reject

(similar to last, must partition each possible way to check)

L^* is decidable

Intersection:

let L_1 and L_2 be decidable languages
TMs M_1 and M_2 decide them
let M decide $L_1 \cap L_2$

on input ω:
① Run M_1 on ω → if rejects, REJECT
② if doesnt reject, run M_2 on ω
if M_2 accepts → ACCEPT
otherwise → REJECT

$L_1 \cap L_2$ is decidable

Complement:

let language L be decidable
TM M_1 decides L
TM M decides $\overline{L}$

on input ω:
① Run M_1 on ω → if rejects, ACCEPT
otherwise → REJECT

$\overline{L}$ is decidable

(if one TM of L rejects it means ω is $\overline{L}$.)

The class of TM-recognizable languages is closed under...

Union:

let L_1 and L_2 be recognizable languages
TMs M_1 and M_2 recognize them
TM M recognizes $L_1 \cup L_2$

On input ω:

1. Alternate running M_1 and M_2 on ω each step
2. if M_1 or M_2 accept $\longrightarrow$ ACCEPT
   if M_1 and M_2 reject/halt $\rightarrow$ REJECT
   (looping) $\rightarrow$ then M loops

M accepts when it reaches Accept state after all the steps. $\rightarrow$ $L_1 \cup L_2$ is Recognizable
M rejects when both M_1 and M_2 reject

Concatenation:

let L_1 and L_2 be recognizable languages
TMs M_1 and M_2 recognize them
TM M recognizes $L_1 L_2$

On input ω:

1. Partition ω into 2 parts ω_1 and ω_2
2. Run M_1 on ω_1 $\longrightarrow$ if halt/reject, REJECT
3. Run M_2 on ω_2 $\longrightarrow$ if accept, ACCEPT
   otherwise $\longrightarrow$ REJECT

$L_1 L_2$ is recognizable

Star:

let L be a recognizable language
TM M_1 recognizes L
TM M recognizes L^*

On input ω:

1. Partition ω into parts $\omega_1 \omega_2 \ldots \omega_n$
2. Run M_1 on ω_i (all parts)
   if M_1 accepts ALL ω_i $\rightarrow$ ACCEPT
   if M_1 halt/rejects any ω_i $\rightarrow$ REJECT
   (must try different partitions)

if $\omega_i \in L$ then there exists a TM M that recognizes L^*

Intersection:

let L_1 and L_2 be recognizable languages
TMs M_1 and M_2 recognize them
let TM M recognize $L_1 \cap L_2$

On input ω:

1. Run M_1 on ω. $\rightarrow$ if halt/reject, REJECT
2. Then Run M_2 on ω $\rightarrow$ if halt/reject, REJECT
   otherwise $\longrightarrow$ ACCEPT

if M_1 and M_2 accept ω $\rightarrow$ M accepts ω
$\omega \in L_1 \cap L_2$

$L_1 \cap L_2$ is recognizable

Complement:

if L and $\bar{L}$ are TM-recognizable
the L is decidable.
from theorem, ~~if L and $\bar{L}$~~
A lang is Decidable
iff both it and its complement are recognizable
TM would NOT Halt for strings not in the language

Problem 4

Proposition: Every infinite TM-recognizable language has an infinite Decidable subset.

let L be an infinite Recognizable language and has an infinite decidable subset.
There exists an enumerator E for all strings in L

let $L' = \{w_1, w_2, w_3, \dots w_n\}$ $i > 1$
w_i is next string in order after w_{i-1}

Consider:

① L' is Infinite and $L' \subseteq L$

let L' be finite
→ E generates all strings $< w_i$, one largest in L'
→ because E generates finite # strings $< w_i$
L is finite → CONTRADICTION!

Also: M only ACCEPTS if in Stnd Order

→* L' is NOT finite → L' is Infinite
Because E. printed all strings $\in L'$,
→* $L' \subseteq L$

② L' is decidable

Can construct E'
on no input:
① prints out strings accepted by M
→ Meaning all in Standard Order
Because L is infinite, E' is infinitely
E' is decidable because prints Stnd order | enumerating
→ Therefore, the language L', of E' is decidable

→ L' is an infinite decidable subset of L, an infinite Recognizable language