1) 1.

$EQ_{DN} = \{ \langle D, N \rangle \mid D$ is a DFA and N is a NFA $L(D) = L(N) \}$

If the input is not the form of $\langle D, N \rangle$

I should reject the input

Otherwise, the input is the form of $\langle R, W \rangle$

N is a NFA. run convertND on $\langle N \rangle$

convertRN will accept on top of $\langle A \rangle$.

A is the DFA of $\langle A \rangle$

run EQ_{DFA} on $\langle D, A \rangle$

If EQ_{DFA} accept $\langle D, A \rangle$, then accepts $\langle D, N \rangle$, otherwise rejects input

Therefore, the language is decidable

2) $L = \{ \langle A, B \rangle \mid A \text{ and } B \text{ are DFAs and } L(A) \subseteq L(B) \}$

If the input is not the form of $\langle A, B \rangle$

I should reject the input

Otherwise, the input is the form of $\langle A, B \rangle$, design

DFAC to check $L(A) \subseteq L(B)$ $L(C) = L(B) \cap \overline{L(A)}$

Determine whether $L(C)$ is the empty language

If it is, then $L(A) \subseteq L(B)$, and the Turing machine accepts; otherwise, it rejects

Therefore, L is decidable

3) $A_{rex} = \{\langle R, W \rangle \mid R$ is a regular expression describing a language over Σ containing at least one string W that has 00 as a substring $\}$

If the input is not the form of $\langle R, W \rangle$)

I should reject the input

otherwise, construct a regular expression T that accept every string that contains the substring 00. run EQ_{RE} on $\langle T, R \rangle$ to test whether $L(T) = L(R)$,

If $L(T) = L(R)$ and W accept by $L(T)$ then accept the input, then reject,

Therefore, A is decidable.

2) 1. $A_{VK} = \{\langle M, K \rangle \mid M$ is a turing machine, and it accepts all strings of length less than $K\}$

If the input is not the form of $\langle M, K \rangle \}$)

I should reject the input

Otherwise, the input is the form of $\langle M, K \rangle$

run M on K.

if it accept all string of length less K,

it accepts the input, otherwise reject