1) 1) $H = \{\langle M, x\rangle \mid M$ is a TM that halts on $x\}$

H is turing recognizable because it is recognized by the following TM:

Turing machine T takes in inputs $\langle M, x\rangle$ where M is a TM and x is its input string. We then run M on x,

if M halts the we accept
else reject

2) $H = \{\langle M, x\rangle \mid M$ is a TM that halts on $x\}$
E = Enumerator that enumerates H
M_i = i^{th} output of E
$S_1, S_2, S_3, S_4, S_5 \dots S_K$ be input strings for M
We will define a Turing machine T as:
$W = S_i$ (w = input)
Enumerate M on E
Run M_i on W
If M_i accepts then reject
If M_i doesnt accept the accept

Since if we run T on itself we get results opposite to how we defined T, H is not decidable,

3) Languages are decidable if and only if it and its complement are recognizable. Since H is recognizable & H is not decidable the $\overline{H}$ is not recognizable

2) 1) $H_0 = \{ \langle M \rangle \mid M \text{ is a TM that halts on } 0 \}$

H_0 is turing recognizable because it can be recognized by the following TM:

Turing machine T takes in input M where M is a TM.
If M halts on input 0 then accept
else,
reject

2) E = Enumerator that enumerates H_0
M_i = i^{th} output of E
input string s_i
if $0 \in S_i$ then reject
Define Turing machine T as
  $w = S_i$
  Enumerate M on E
  Run M_i on w
  If M_i accepts then reject
  If M_i doesnt accept then accept

If we run $T\langle T \rangle$ we will get results opposite to how we defined T above, H_0 must not be decidable.

3) Since a language is decidable if both it & its complement are recognizable since H_0 is recognizable & not decidable then $\overline{H_0}$ must be not recognizable.