solution for Oct. 17 Daily Question

Construct a PDA that recognizes the language over $\{a,b\}^*$

$$\{w \mid \exists\, x.\ xx^R = w\}$$

where x^R denotes the reverse of x, as usual.

Let the PDA $P = (Q, \Sigma, \Gamma, \delta, q, \perp, F)$, where

- $Q = \{q, r\}$, the set of states of the NFA
- $\Sigma = \{a,b\}$, the alphabet of the NFA
- $\Gamma = \{a, b, \perp\}$, the alphabet of the stack
- q = the initial state of the NFA
- $\perp$ = the empty-stack symbol
- F – the PDA accepts on an empty stack
- $\delta = \{$ $(q, \varepsilon, \perp, r, \varepsilon)$,
  $(q, a, \perp, q, a\perp)$,
  $(q, b, \perp, q, b\perp)$,
  (q, a, a, q, aa),
  $(q, a, a, r, \varepsilon)$,
  (q, b, a, q, ba),
  (q, b, b, q, bb),
  $(q, b, b, r, \varepsilon)$,
  (q, a, b, q, ab),
  $(r, a, a, r, \varepsilon)$,
  $(r, b, b, r, \varepsilon)$,
  $(r, \varepsilon, \perp, r, \varepsilon)$ $\}$

The machine works in such a way that it adds symbols in the first half of the string to the stack, while remaining in state q to represent the fact that it is looking at the first half. It makes a nondeterministic guess as to when it has reached the halfway point of the string, at which point it transitions to state r, and from then on it cannot push new symbols onto the stack, only pop symbols as it matches them to those in the second half.