```
// This struct is the main component of the Enigma machine which combines the other components
//  required to encode or decode messages.
type EnigmaMachine struct {
   plugboard Plugboard
   reflector Reflector
   rotorSet RotorSet
}

// Plugboard allows swapping of characters before they are processed by the rotors. It has a
// ‘connections’ map which maps one character to another, indicating which characters are swapped.
type Plugboard struct {
   connections map[rune]rune
}

// Reflector represents the fixed reflector.
type Reflector struct {
   wiring [26]int
}

// This struct represents a single rotor of the Enigma machine. It has a wiring array which contains the
// mapping of input characters to output characters, a notch rune which indicates when the next rotor
 //should be rotated, a position rune which indicates the current position of the rotor.
type Rotor struct {
   wiring []rune
   notch rune
   position rune
   ringSetting rune
   turnover rune
   turnoverFlag bool
}
// RotorSet represents the set of three rotors that make up the rotor mechanism of the machine.
type RotorSet struct {
    leftRotor Rotor
    middleRotor Rotor
    rightRotor Rotor
}

// InputRotor represents a rotor that is used for input. Also, it is used to allow the operator to specify the
// initial state of the rotors before encoding or decoding a message.
type InputRotor struct {
    wiring []rune
    position rune
}
```