1. Write a `pois.prob()` function that computes $P(X = x)$, $P(X \neq x)$, $P(X < x)$, $P(X \leq x)$, $P(X > x)$, and $P(X \geq x)$. Enable the user to specify the rate parameter λ.

```
pois.prob <- function(x, lambda, type="<="){
  # Use dpois and ppois to conditionally return the correct probability
  if (type == "="){
    return(dpois(x, lambda))
  }
  else if (type == "!="){
    return(1 - dpois(x, lambda))
  }
  else if (type == "<"){
    return(ppois(x - 1, lambda))
  }
  else if (type == "<="){
    return(ppois(x, lambda))
  }
  else if (type == ">"){
    return(1 - ppois(x, lambda))
  }
  else if (type == ">="){
    return(1 - ppois(x - 1, lambda))
  }
}
```

2. Write a `beta.prob()` function that computes $P(X = x)$, $P(X \neq x)$, $P(X < x)$, $P(X \leq x)$, $P(X > x)$, and $P(X \geq x)$ for a beta distribution. Enable the user to specify the shape parameters α and β.

```
beta.prob <- function(x, alpha, beta, type="<="){
  # Use dbeta and pbeta to conditionally return the correct probability

  if (type == "="){
    return(dbeta(x, alpha, beta))
  }
  else if (type == "!=") {
    return (1 - dbeta(x, alpha, beta))
  }
  else if (type %in% c("<", "<=")){
    return(pbeta(x, alpha, beta, lower.tail = TRUE))
  }
  else if (type %in% c(">", ">=")){
    return(pbeta(x, alpha, beta, lower.tail = FALSE))
  }
}
```

The `pois.prob()` and `beta.prob()` functions compute probabilities for the Poisson and Beta distributions, respectively. The `pois.prob()` function calculates the probabilities for the Poisson-distributed random variable with a given rate parameter (λ), using `dpois()` for exact probabilities and `ppois()` for cumulative probabilities. The `beta.prob()` function does the same for a Beta-distributed variable with shape parameteres (α, β) using `dbeta()` for density values and `pbeta()` for cumulative probabilities.