```
scala> def squareRoot(n: BigDecimal): Stream[BigDecimal] =
     |   {
     |     def squareRoot(guess: BigDecimal, n: BigDecimal): Stream[BigDecimal] = {
     |       Stream.cons(guess, squareRoot(0.5 * (guess + n / guess), n))
     |     }
     |     squareRoot(1, n)
     |   }
squareRoot: (n: BigDecimal)Stream[BigDecimal]

scala>

scala>   squareRoot(2)
res10: Stream[BigDecimal] = Stream(1, ?)

scala>

scala>   val iterations = 5
iterations: Int = 5

scala>

scala>   squareRoot(2)(iterations - 1)
res11: BigDecimal = 1.41421356237468991062629557889013 5

scala>

scala>   squareRoot(2).take(iterations).toList
res12: List[BigDecimal] = List(1, 1.5, 1.41666666666666666666666666666666, 1.41421568627450980392156862745098039215686274509803921568627450980, 1.41421356237468991062629557889013 5)

scala>
```