

Projeto P2 - Programação Funcional

Aluno: Tarcísio Deschamps Silva

A) Função nOcupados passando como parâmetro qual o numero da fila que quer saber quais cadeiras estão ocupadas.

```
object estadio {
  class estadioA{
    var fila:Map[Int, Array[Boolean]] = Map()
    def nFilas{ println(fila.size + " filas")}
    def nOcupadas(filan:Int){
      println(this.fila(filan).length + " cadeiras na fila " + filan)//imprime a
      quantidade de cadeiras na fila
      var ocupados = 0
      for (x <- this.fila(filan)){
        if(x == true){ocupados+=1}
      }
      println(      + " cadeiras ocupadas")
    }
  }
}

def main(args: Array[String]): Unit = {
  var s1= new estadioA;
  s1.fila += (1 -> Array(true, false, false, true)) //fila 1
  s1.fila += (2 -> Array(false, false, true, true, false, true)) //fila 2
  s1.nFilas//Imprime a quantidade de filas
  println(s1.nOcupadas(2)) //imprime ocupados da fila 2
}

}
```

C) Usando o código da questão A com o incremento abaixo, chamar no main

```
println(s1.ocupadas(2, 6))
```

para pegar o elemento daquele setor s1, na fila 2 e cadeira 6, retornando resposta em boolean

```
def ocupadas(filan:Int, cadeiraN:Int){
  println("cadeira " + cadeiraN+ " da fila "+filan+" está ocupada?")
  println(fila(filan)(cadeiraN-1)) // usado o '-1' para não ter que nomear a
  cadeira pelos indices e sim por numeros, iniciando por 1
  |
}
```

E) Incremento na classe dos códigos acima e chamando no main println(s1.maiorQtVagas)

```
def maiorQtVagas{
  var i=1
  var max =
  while (i< this.fila.size+1){
    println(nOcupadas(i)) //chama nOcupadas para ver ocupadas de todas as filas
    if(fila(i).size > max) max = fila(i).size
    i=i+1 //incrementa para percorrer outras filas
  }
  println("maior fila contem: "+max);
}
```