Antonio Gallego Peñalver

Ejercicios avanzados Pyspark - completar map y reduce

Map

```
    def bfsMap(node):
        heroID = node[0]
        data = node[1]
        connections = data[0]
        distance = data[1]
        status = data[2]

        results = []

        if status == 'Doing':
            for connectionHero in connections:
                # Añade un nodo por cada conexión
                results.append((connectionHero, ([], distance + 1, 'Doing')))


                # Si se alcanza el nodo buscado, se incrementa el contador
                if connectionHero == targetHeroID:
                    hitCounter.add(1)

            # Añade el nodo recibido como procesado
            results.append((heroID, (connections, distance, 'Done')))

        else:
            # Añade el nodo recibido sin procesar
            results.append((heroID, (connections, distance, status)))

        return results
```

Reduce

```
def bfsReduce(data1, data2):
    connections1 = data1[0]
    connections2 = data2[0]
    distance1 = data1[1]
    distance2 = data2[1]
    status1 = data1[2]
    status2 = data2[2]

    # Unifica las dos listas de conexiones
    connections = list(set(connections1 + connections2))

    # Se queda con la distancia menor
    distance = min(distance1, distance2)

    # Se queda con el estado más avanzado
    status_priority = {'N/A': 0, 'Pend': 1, 'Doing': 2, 'Done': 3}
    status = status1 if status_priority[status1] > status_priority[status2] else status2

    return (connections, distance, status)
```

Antonio Gallego Peñalver

Programa principal

```
iterationRdd = createStartingRdd()

iterationRdd.collect()

for iteration in range(1, 10):
    print("Procesando iteración # " + str(iteration))

    # Expande nodos Doing, generando registros para sus vecinos con la distancia incrementada. El nodo expandido se añade como finalizado.
    # El resto de nodos se queda igual.
    # Si se alcanza el nodo buscado, se incrementa el acumulador para indicar que hemos terminado.

    mapped = iterationRdd.flatMap(bfsMap)

    # Se ejecuta la acción mapped.count() para forzar la evaluación del RDD y la actualización del acumulador
    print("Procesando " + str(mapped.count()) + " valores.")

    if (hitCounter.value > 0):
        print("Se ha localizado el objetivo. Ramas paralelas en las que se ha alcanzado: " + str(hitCounter.value))
        break

    # Reducer combina registros de cada id, uniendo los nodos adyacentes, y dejando el número de pasos menor y el estado más avanzado
    iterationRdd = mapped.reduceByKey(bfsReduce)
```

▸ (3) Spark Jobs

```
Procesando iteración # 1
Procesando 8330 valores.
Procesando iteración # 2
Procesando 220615 valores.
Se ha localizado el objetivo. Ramas paralelas en las que se ha alcanzado: 1
```