# TP4 SparkStreaming

ssh -i UE-INF2479M\_Cle\_SSH\_TP3\_p1501263.pem p1501263@192.168.76.141

cp /mnt/c/Users/etulyon1/Downloads/tortoiseListener.py .

scp -i UE-INF2479M\_Cle\_SSH\_TP3\_p1501263.pem tortoiseListener.py p1501263@192.168.76.141:/home/p1501263/tp4

spark-submit tortoiseListener.py > res\_tortoiseListener.log

(timestamp, id, top, position, nbAvant, nbTour)

#### filtrer que les tuples qui concerne une tortue. Pour cela, vous considérez comme votre tortue, celle dont l'id correspond à votre numéro de groupe (colonne Num\_equipe\_tortues dans Tomuss)

tortoises = tortoises.filter(lambda x: len(x)>1 and x[1] == "4")

#### convertir les données du flux en objet JSON respectant le schéma (id, top, position, nbAvant, nbTotal)

def convert\_to\_json(record):

return {

"id": record[1],

"top": record[2],

"position": record[3],

"nbAvant": record[4],

"nbTour": record[5]

}

json\_data = tortoises.map(convert\_to\_json)

json\_data.pprint()

#### connaitre la distance parcourue par votre tortue. Le résultat sera exprimé en nombre de cellules. Vous veillerez à tenir compte des tour complet

"distance": int(record[5])\*tailleParcours + int(record[3])

#### connaitre le podium individuel et temporaire (les tortues qui sont sur les 3 premières marches du podium; attention, il peut y avoir des ex aequo) tous les 30 secondes. Le résultat sera un objet JSON de la forme {'1er': [id1, id2], '2eme':[id3], '3eme:[id4, id5]}.