
# School of Computer Science

# COMP47470

# Project 3
# Graph Processing

| | |
|---|---|
| **Teaching Assistant:** | Patrick Cormac English |
| **Coordinator:** | Dr Anthony Ventresque |
| **Date:** | Wednesday 14th April, 2021 |
| **Total Number of Pages:** | 3 |

## General Instructions

- Some of the following problems may require a sequence of operations to solve - it is not necessary to design a single operation that will generate the answer, although you should provide an answer that best represents your learning to date - as an example, a very inefficient answer will be acceptable, but may not receive the same grade as an answer that creates a solution in half the operations!
- This project is largely based on the material covered in the lab sessions. However, you may desire additional functionality (such as Java string manipulation functions for Hadoop). We would encourage external research in this regard - the documentation for Scala/Java etc. will be helpful for you. As an example, Scala functionality may be explored **here**. The MapReduce documentation can be found **here** and the latest GraphX docs are **here .**
- We ask you to hand in an archive (zip or tar.gz) of your solution: code/scripts, README.txt file describing how to run your programs, a short pdf report of your work (no need to include your code in it).
- The report should not be longer than 10 pages (this is not a hard constraint though).
- The report must be submitted as a PDF, but stylistically you may use whichever software you like to prepare it (e.g. Word, Google Docs, Latek) as long as any code included can be copy-pasted!
- For Sections 1 and 2, each answer in the report should include the following:
  - The answer to the question (usually a number)
  - The code you used - either include a snippet (if the code is short) or reference an attached script/txt file. You may use scala scripting for this assignment, but a text file including **all lines of code you used (NB - running them sequentially must replicate your work)** will be accepted.
  - A brief explanation of what your code is doing/how it operates. This can be very short (one sentence) as long as it demonstrates an understanding of each portion of your code. For extensive scala commands (e.g. with multiple functions strung together) please make sure you explain what each portion is doing.
- The breakdown of marks for the project will be as follows:
  - Exercise 1 (Hadoop Graph Processing): 30%
  - Exercise 2 (GraphX): 40%
  - Exercise 3 (Reflection): 30%
- **Due date: 05/05/2021**

# 1 Hadoop Graph Processing

In this section you will run Hadoop MapReduce jobs on a dataset of actors and films including the actor Kevin Bacon (you can read more about Mr.Bacon and the "Bacon Number" phenomenon here). You may use the Hadoop environment from previous projects/labs. The data includes a unique integer value for each actor and film that you may find useful. Given the "pair" nature of the data you may find it useful to look at the "dictionary" data structure in java: **here**. Download the data using the following code (remember to make it all one line!):

```
wget --no-check-certificate
↪ 'https://docs.google.com/uc?export=download&id=1bfv8UeEVb8ciiQU8Df7I4o727ImaSWrf'
↪ -O bacon.csv
```

Using a MapReduce script, complete the following graph exploration tasks. You may "merge" the outputs of multiple MapReduce tasks with further MapReduce tasks. You may also create transformed outputs using MapReduce that generate your final answers (e.g. your final output does not need to be generated from the original bacon.csv file). Remember - your MapReduce tasks can take multiple inputs! You may present multiple mapreduce outputs as your answer, if the answer is split across them (e.g. for Q4 below - you may find A-B and B-C separately, although a better answer might have them in a single output).

1. Create a list of the number of actors that appear in each film
2. Peter Cushing appears in only one film - what is it?
3. What film connects Cate Blanchett and Christina Ricci (i.e. what film did they both act in).
4. What is the path between Audrey Gelfund and Kevin Bacon (This will take the form Audrey Gelfund - ? - John Malkovich - ? - Kevin Bacon) where ? stands for a single film.
5. In this dataset, 3 actors have a bacon number of 1 - they have acted in a film with Kevin Bacon. Identify these actors. As a hint, consider how you could compare two MapReduce outputs with different Map conditions to find the films that star more than just Kevin in this dataset (it's only 4!)

# 2 GraphX

For this section, you will be assessing the same data as above, but we have also provided a mirrored version to allow you to treat this as an undirected graph. You may use either version of the data as appropriate. The following documentation will be very useful for you: **GraphOps Documentation**. For all of the following, your answer need not be a single command - you may create outputs and perform follow-up operations on those outputs, but GraphX should be the "driver" in your answers - e.g. an answer that exclusively uses SparkSQL unions (and no GraphX operations) to make links would be undesirable. Download the data using the following code (remember to make it all one line!):

```
wget --no-check-certificate
↪ 'https://docs.google.com/uc?export=download&id=1YLLa_PTAa3o-fnkKptg7socBLv-ww6kI'
↪ -O bacon_mirror.csv
```

1. Import the data and create a graph representing the data
2. How many nodes and vertices are there in the graph
3. Which films star more than 2 actors in this dataset (i.e. which film are connected to more than one actor node in the graph). As a hint, consider how you could apply last project's counting tasks to the output of a Graph operation!
4. What films has Kevin Bacon starred in?
5. Which actor has starred in the most films?

## 3 Reflection

Write a short report (max. 1 page) on the following question: how would one use MapReduce and GraphX to generate a list of second neighbours (i.e node C from A in A-B-C)? What tools could you use, and what information would you need to have. Identify difficulties that might arise using both tools, and how you might address them.