ENSF 612: Midterm Fall 2021

Marks: 40

Duration: 48 hours

The midterm is worth 25% of course marks

Instructions:

1. This midterm is an open book examination.
2. It must be completed individually. You cannot copy from others or consult with others. Any identification of such unauthorized actions may result in 0 grade for this assignment.
3. This is a take home examination.
4. Each student will submit his/her solution in a PDF file and upload it in D2L under Midterm Folder.

Question 1: (Marks 10)

a) State True or False. "With a Breadth-First Search, the left node contains the smaller of two values between the left and right nodes in a given level" (Marks 1)

b) State True or False. "With a Distributed File System, multiple copies of the same chunk of data could be placed on the same chunk server" (Marks 1)

c) Briefly state how a map-reduce platform recovers from a reduce worker failure. (Marks 2)

d) A map-reduce program employs 5 reducers with ids 0, 1, 2, 3, and 4. The platform generates a hash code of 31 for one of the keys emitted by a mapper. Which reducer will get that key? (Marks 1)

e) Briefly state how you can implement Breadth-first-search algorithm in a map-reduce platform. Assume you have the following data in binary tree [20, 30, 10, 50, 60, 100, 90]. Explain how you can use the algorithm in the map-reduce platform to search for a value in the tree like 60. (Marks 5)

Question 2: (Marks 10)

You are given the records of an issue tracking system in a software system. Each recorded issue is of the form <IssueId, CreationTime, IssueDescription, IssueSeverity, NumberOfComponentsAffected>. The columns are described below.

1. IssueId = the issue Id (a numeric Id)
2. CreationTime = the time when the issue was logged into the system
3. IssueDescription = the textual contents of the issue (e.g., database B is down)
4. IssuePriority = a value between 1 and 5 (5 means the highest priority)
5. NumberOfComponentsAffected = Total number of software components affected by the issue (e.g., website1, website2 using database B are affected)

This question is divided into three tasks below.

Task 2.1 (Marks 2).

Design a Spark transformation analytics that will add the following additional columns per review.

1. IssueType (e.g., an IssyeType can be bug, feature, etc.)
2. IssueYear
3. IssueMonth
4. IssueDay

Assume that you have access to the following function that you can access via Spark

1. getIssueType(IssueDescription) will return 'b' for bug, 'f' for new feature and 'e' for feature enhancement – the types are automatically determined by analyzing the IssueDescription automatically
2. getYear(CreationTime) will return year of the CreationTime
3. getMonth(CreationTime) will return month of the ReviewTime
4. getDay(CreationTime) will return day of a week (e.g., Monday, Tuesday, etc.) of the CreationTime

Task 2.2 (Marks 3).

Design a Spark transformation action that will return the total number of components affected by all the issues

Task 2.3 (Marks 5).

Design Spark transformation action, one each for the following requirement:

1. The average overall IssuePriority per issue (Mark 1)
2. The total number of issues by each issue type
3. The total number of issues reported by
    a. Each year
    b. Each month
    c. Each day

Question 3: (Marks 10)

You are given Web server log records of type <***Access_time, Client_IP, URL_requested, Size_of_data_transferred***>. ***Access_time*** is the time at which a client's request was received. ***Client_IP*** denotes the IP address of the client issuing the request. ***URL_requested*** is the name of the URL requested by the client while ***Size_of_data_transferred*** is the size in bytes of the response transferred from the server to the client.

You are asked to write a map-reduce analytic that outputs records of type <***Client_country, Client_city, Total_size_of_data_transferred***>. The first two fields represent the country and city to which the ***Client_IP*** is assigned. ***Total_size_of_data_transferred*** represents the total size of data transferred by the server to IP addresses belonging to the ***Client_city.*** The **records belonging to the same country should appear in the output of the same reducer.** Your solution should involve a single map-reduce stage with multiple reducers. Assume that you have appropriate

function(s) to obtain ***Client_country*** and ***Client_city*** given ***Client_IP***. Clearly sketch out your solution specifying the following:

- details about the combiner and partitioner (if used and applicable)
- the input record to a map call and the output(s) generated by the map call
- the input record to a reduce call and the output generated by the reduce call
- pseudo code for map, combine (if applicable) and reduce and command line options for the partitioner (if applicable)

Question 4: (Marks 10)

You are given records of the following format pertaining to a large social network:

Profile_id, <Friend_profile_id_list>

Profile_id uniquely identifies a member. <***Friend_profile_id_list***> is a list containing the ***Profile_id***s of all the friends of this member.

1. Provide pseudo code of the above solution in Pyspark.
    a. The count of total friends of a given Profile_id
    b. The list of common friends between any pair of fiends in the network (assume a pair as two friends with ID, Profie_id1, Profile_id2).
2. Describe how the above network can be implemented using a map-reduce program that outputs for each Profile_id the total number of friends. Clearly state the key-value pairs involved and clearly describe how the map and reduce transform their inputs. **Your solution should only involve a single map-reduce stage.**