# COP 5536 Fall 2016

## Programming Project

Due Date: Nov 18th, 2016, 11:55 pm EST

## 1. General

### Problem description

You are required to implement a system to find the n most popular hashtags appeared on social media such as Facebook or Twitter. For the scope of this project hashtags will be given from an input file.
Basic idea for the implementation is to use a max priority structure to find out the most popular hashtags.

You must use following structures for the implementation.

1. Max Fibonacci heap: use to keep track of the frequencies of hashtags.

2. Hash table: Key for the hash table is hashtag and value is pointer to the corresponding node in the Fibonacci heap.

You can assume there will be a large number of hashtags appears in the stream and you need to perform increase key operation many times. Max Fibonacci heap is required because it has better theoretical bounds for increase key operation.

### Programming Environment

You may implement this assignment in Java or C++. Your program must be compilable with the compilers and run on the following environment:
- Java or g++/gcc on the thunder.cise.ufl.edu server
You may access the server using Telnet or SSH client on thunder.cise.ufl.edu.

You must write a makefile document which creates a executable file named hashtagcounter.

## 2. Input and Output Requirements

Your program is required to take the input file name as an argument. Following is an example of a program that read from a file named file_name.

```
$hashtagcounter file_name
```

### <u>Input Format</u>

Hashtags appear one per each line in the input file and starts with # sign. After the hashtag an integer will appear and that is the count of the hashtag (There is a space between hashtag and the integer). You need to increment the hashtag by that count. Queries will also be appeared in the input file and once a query appears you should append the output to the output file. Query will be an integer number (n) without #

sign in the beginning. You should write the top most n hashtags to the output file. Once it reads the stop (without hashtag) program should end. Following is an example of an input file.

```
#saturday 5
#sunday 3
#saturday 10
#monday 2
#reading 4
#playing_games 2
#saturday 6
#sunday 8
#friday 2
#tuesday 2
#saturday 12
#sunday 11
#monday 6
3
#saturday 12
#monday 2
#stop 3
#playing 4
#reading 15
#drawing 3
#friday 12
#school 6
#class 5
5
stop
```

## **Output Format**

Once a query is appeared you need to write down the most popular hashtags of appeared number in to the output file in descending order. Output for a query should be comma separated list without any new lines. Once the output for a query is finished you should put a new line to write the output for another query. You should produce all the outputs in the output file named “output_file.txt”.

Following is the output file for the above input file.

```
saturday,sunday,monday
saturday,sunday,reading,friday,monday
```

## **You can have following assumptions**

1. No uppercase letters in the input stream
2. No spaces in the hashtags. (only one word per one hashtag)
3. For two hashtags to be same, whole word should match. i.e. #playing and #playing_games are two different hashtags.
4. One query has only one integer.
5. If there are more than one hashtag with similar frequencies, you can print them in any order.

6. Query integer, n<=20. i.e. you need to find at most 20 popular hashtags.
7. Input file may consist of more than 1 million hashtags.

In Fibonacci heaps removeMax() will remove the hashtag from the heap. So you need to re-insert them after printing in order make sure they are still in the heap.

## 3. Submission

The following contents are required for submission:

1.Makefile: Your make file must be directly under the zip folder. No nested directories.

2. Source Program: Provide comments.

3. REPORT:
· The report should be in PDF format.
· The report should contain your basic info: Name, UFID and UF Email account
· Present function prototypes showing the structure of your programs. Include the structure of your program.

To submit, please compress all your files together using a zip utility and submit to the canvas system. You should look for Assignment Project for the submission.
Your submission should be named ***LastName_FirstName.zip.***

Please make sure the name you provided is the same as the same that appears on the Canvas system. Please do not submit directly to a TA. ***All email submission will be ignored without further notification. Please note that the due day is a hard deadline. No late submission will be allowed. Any submission after the deadline will not be accepted***.

## 4. Grading Policy

Grading will be based on the correctness and efficiency of algorithms. Below are some details of the grading policy.
Correct implementation and execution: 60%
Comments and readability: 15%
Report: 25%
Note: ***If you do not follow the input/output or submission requirements above, 25% of your score will be deduced.*** In addition, we may ask you to demonstrate your projects.

## 5. Miscellaneous

- **Do not use complex data structures provided by programming languages**. You have to implement Max-Fibonacci heap data structure by your own using primitive data structures such as pointers. But if you want you can use a library implementation for the hashtable.

- Your implementation should be your own. **You have to work by yourself for this assignment** (discussion is allowed). Program code will be checked and you may have negative result if it has reasonable doubt.