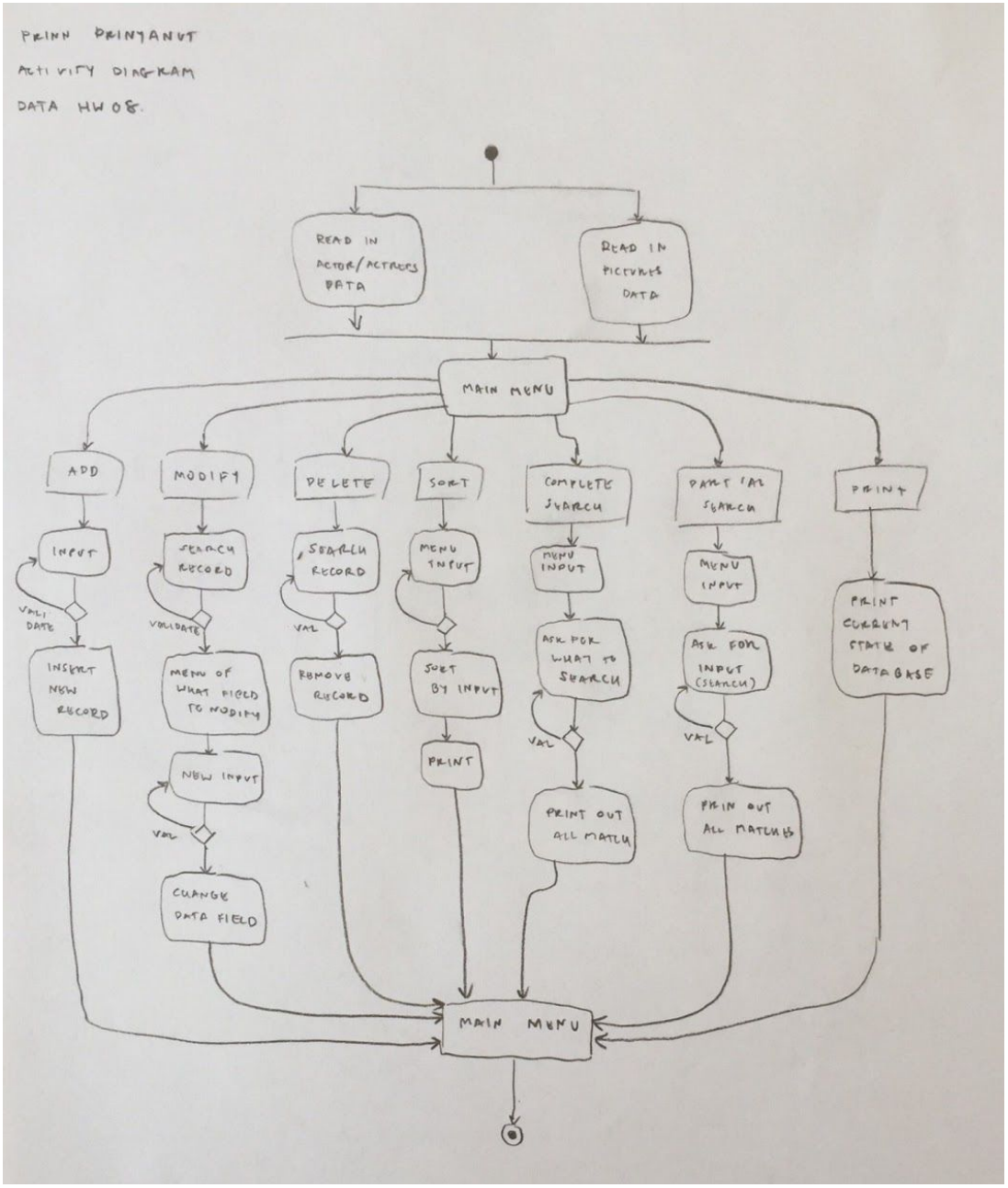
Project Design/Implementation Document (Final Project)

1. Title
    - Oscar Database
    - Prinn Prinyanut
    - May 2, 2018

2. Problem Description
    - Build a simple database system that handle multiple records. This database will be able to store information from files, adding and deleting records, change record fields, allow user to sort records based on keys, and print out the updated database.

3. Overall Software Architecture
    - This program will allow users to manipulate datas in oscar's database. First, the program will read in data from actor/actress.csv and pictures.csv and store them in a BST sorted by name. Then it will take the users to main menu that asks for which database they want to modify (actor/actress or pictures). In each database menu, users will be able to add records, change records, delete records, sort records by users input, do a complete search, do a partial search, and print out current database version.

Activity Diagram



- When first run the program both actor/actress and pictures files will be read in and store in BST.
- Then main menu will appear with 3 options
    i. actor/actress database
    ii. Pictures database
    iii. Nominations database (extra credit)

- Actor and Pictures database will have 7 options to interact with itself.
    i. Add records
        1. Ask for new data for new record
    ii. Modify records
        1. Search for record
        2. Menu of what field to modify
        3. Enter new data
    iii. Delete records
        1. Search for record
        2. Delete record
    iv. Sort records
        1. Ask for which field user wants to sort records by
        2. Print updated records
    v. Complete search
        1. Ask for which field user wants to search
        2. Ask what to search
        3. Print found searches
    vi. Partial search
        1. Ask for which field user wants to search
        2. Ask what to search
        3. Print found searches
    vii. Print records
        1. Print all current state of the database
- Nominations database will have 4 options
    i. Print winner with a given year
    ii. Print all Actor in leading role who won
    iii. Print all Actor or Actress has more nominations in total
    iv. Print all films that starts with 'The'
- After each task is finished user will be guided back to the main menu

4. Input Requirements

- Main menu
    i. Ask if they want to manipulate data for actor/actress database or pictures database (integers, switch case)
- Database menu
    i. Asks what function do they want to do (integers, switch case)
- Add records function
    i. User will be able to add info using strings and integers
- Modify records
    i. Input to search record by name (strings)
    ii. Menu of what field they want to modify (integers, switch case)

        iii. New data for that specific field (strings or integers)
    - Deleting records
        i. Input to search record by name (strings) that want to delete
    - Sort by input
        i. Menu that allow user to sort the based on different field (integers, switch case)
    - Complete Search
        i. Menu that allow user to sort the based on different field (integers, switch case)
        ii. Ask for input to do a complete search (strings / integers / double)
    - Partial Search
        i. Menu that allow user to sort the based on different field (integers, switch case)
        ii. Ask for input to do a partial search (strings / integers / double)
    - Nomination extra credit
        i. Print winner with a given year
            1. User need to enter year

5. Output Requirements
    - Menus
        i. Using switch case (integers)
    - Print out tree after add
    - Print out tree after modify
    - Print out tree after delete
    - Print out tree after sorted
    - Print out complete search found results
    - Print out partial search found results
    - Print current tree and store them in an output file

6. Problem Solution Discussion
    - In this project we have to create databases that allow users to interact with it. When start the program user will be presented with 3 databases. One for actor/actress, pictures and nominations. Then for actor and pictures databases user will be able to add records, modify records, delete records, sort records by different fields, do a complete search for records, do a partial search for records and print records. Nomination database will print out interesting datas. The main data structure that will be used in this program is BST, which is where we will be storing all of the data.

7. Data Structures
    - Storing results for complete and partial search
        i. When search is found I will be storing that data in a vector. The reason why I chose vector over other data structure in this case is because all I'm doing in this function is just storing the found data and printing it later. Also vectors had methods such as .push_back and .size that will allow me to keep track of how many searches that were found. The complexity is O(n). I know that this is not the most efficient way to store found data, but compare to linklist, search is still O(n). I'm more comfortable with vector that's why I choose vector.
    - Storing results for second complete and partial search
        i. For the second search I will be searching through the first found vector. I thought that this makes the most sense because search is still O(n).
    - Sorting different fields
        i. Every time user wants to sort by different field I will be copying the entire data into a new sorted BST.

8. User Interface Scheme
    - When user first run the program, they will land on a home page. They will see a welcome message and 3 options "actor/actress database", "pictures database", and "nominations database". Then in each database user will be presented with another menu that allows them to interact with the database. For actor and pictures they will be able to add, modify, delete, sort, complete search, partial search, print records, and back to main menu. For nominations user will be able to print winner with a given year, print all Actor in leading role who won, print all Actor or Actress has more nominations in total, print all films that starts with 'The', and back to main menu
    - Ex:
    - Welcome!
    - actor/actress database
        i. Add records
        ii. Modify records
            1. Search by name
            2. Menu asking which field to modify
        iii. Delete records
            1. Search by name
        iv. Sort records
            1. Menu asking which field to sort by

        v. Complete search records
            1. Menu asking which field to search by
            2. Enter search input
        vi. Partial search records
            1. Menu asking which field to search by
            2. Enter search input
        vii. exit
    - pictures database
        i. Add records
        ii. Modify records
            1. Search by name
            2. Menu asking which field to modify
        iii. Delete records
            1. Search by name
        iv. Sort records
            1. Menu asking which field to sort by
        v. Complete search records
            1. Menu asking which field to search by
            2. Enter search input
        vi. Partial search records
            1. Menu asking which field to search by
            2. Enter search input
        vii. Exit
    - Nominations database
        i. Print winner with a given year
            1. Enter year
        ii. Print all Actor in leading role who won
        iii. Print all Actor or Actress has more nominations in total
        iv. Print all films that starts with 'The'
    - Quit program

9. Status of Application
    - The program runs successfully.
    - I developed and tested on Xcode g++. It was compiled, run, and tested on csegrid.ucdenver.pvt.
    - Extra credit: COMPLETE!