CSS-340

03 November 2020

Pseudocode for files

For the App source file

Pseudocode:Default constructor

```
Apps::Apps(){   // Sets the default values for the private values in the
Apps class
   name;
   category;
   rating;
   installs;
}
```

Pseudocode:Second Default constructor

```
Apps::Apps(string name, string category, double rating, float installs)
{
      Name;
      Category;
      Rating;
      Installs;
]
```

Pseudocode:Getters

```
string Apps::GetName() const{
```

```
            Returns name;
    ]
    string Apps::GetCategory() const{
       returns category;
    }
    double Apps::GetRating() const{
       returns rating;
    }
    float Apps::GetInstalls() const{
       returns installs;
    }
    void Apps::PrintInfo() const {  //Prints the values of the app
       Print out the results and number of ratings
    }
```

Pseudo for Ranking source file:

```
    RankingApp::RankingApp(Apps app)//reference .h file {
    Read app
    }
    Apps RankingApp::GetApp() const{
            Return app;
}
void RankingApp::PrintInfo() const {
```

Declare two string values for reading

Declare two ints one as a temp

Declare a data type of double for rating and installs;

Open your file/directory

Create a loop inside to check the file

Inside the file make a condition to retrieve the category, ranking, and # of installs

The new if statement where the getter rating is equal to the that is requested then make a Nested if statement to compare installations if true then displays rank and close file.

```
{
    if(app.getter==rating)
    if(app.category==instalss)
        if(comparison of category and size )
            move to a temp rank
            close file
    }
Then close the file
Print the ranking info by this case by our limit of 500
Then display what rank it currently is.
}
```

Pseudocode Algorithms:

The sorting algorithm is bubble sort. Wanted to implement but had trouble doing so.

```
void bubbleSort(int arr[], int n)
```

```
{
        int i, j;
        for (i = 0; i < n-1; i++)

        // Last i elements are already in place
        for (j = 0; j < n-i-1; j++)
          if (arr[j] > arr[j+1])
            swap(&arr[j], &arr[j+1]);
      }
```

Built-in datatypes:

In our main we have 3 data types string, double, and int that get called back to be used in the other 2 classes. We have 2 classes, the App header contains the private data members string name,category, double rating, and long long installs. The Ranking header contains the data type int ranking.

User-defined datatypes:

For our user defined datatypes, as I mentioned there are 2 classes both containing individual header files and a source file. In the App header file we have seven public functions. One default constructor and a second constructor to initialize the private fields. Then, we have string GetCategory() const;, double GetRating() const; float GetInstalls() const;void PrintInfo() const. The getters just retrieve the information, but the print function displays the name,rating, and number of installs. For the second header file, Ranking.h, contains RankingApp(Apps app) setting the app and the getter Apps GetApp() const to retrieve the information. Finally the last function in this header is the void PrintInfo() const and inside this function a file was opened making it so that the file could be read. Within that function it searches for the ranking and displays the result.