**Problem 3:**

**Class.h**

#include <iostream>

#include <string>

using namespace std;

class Movie{

private:

string name;

string rating;

int gross;

public:

//getters

string getName(){ return(name); }

string getRating(){ return(rating); }

int getGross(){ return(gross); }

//setters

//pre: Movie m exists

//post: Movie m's name will be set to paramName

void setName(Movie m, string paramName){ m.name = paramName; }

//pre: Movie m exists

//post: Movie m's rating will be set to paramRating

void setRating(Movie m, string paramRating){ m.rating = paramRating; }

/\*void getGross(Movie m, int gr){ m.gross = gr; }\*/

//You shouldn't be able to set the gross of the movie

//given that there is a function that calculates this

double movieShowing(int n){

double revenue = n\*(12.00);

double total = gross + revenue;

gross += revenue;

return total;

}

//Pre: Movie m has a rating

//Post: The ratings will be compared

//This function is an integral part of a larger function to set the rating for a combined movie

string compareRatings(Movie m){

int paramReturnNum = ratingToNum(m.getRating());

int baseReturnNum = ratingToNum(rating);

if(paramReturnNum > baseReturnNum){

return m.getRating();

}

else{

return rating;

}

}

//Pre: Movie m has a rating

//Post: The ratings will be translated into integer values for easier comparison

//This function is an integral part of a larger function to set the rating for a combined movie

int ratingToNum(string rate){

int paramRating;

if(rate == "G"){ paramRating = 1; }

else if(rate == "PG"){ paramRating = 2; }

else if(rate == "PG-13"){ paramRating = 3; }

else{ paramRating = 4; }

return(paramRating);

}

//Pre: Movie m exists

//Post: The two movies will be combined if, and only if, the two movies have the same name

//Otherwise a new empty movie is created.

Movie mergeMovie(Movie m){

Movie combinedMovie;

if(m.name == name){

combinedMovie.rating = compareRatings(m);

combinedMovie.gross = m.gross + gross;

combinedMovie.name = name;

}

else{

combinedMovie.name = "Could not combine.";

}

return combinedMovie;

}

//Default Constructor

//Creates a movie with default variables

Movie(){

name = "noName";

rating = "R";

gross = 0;

}

//Overloaded constructor that will be most commonly used as you create a movie with some sense

//Pre: So long as paramName and paramRating are strings and paramGross is an integer

//Post: A Movie object will be created with the entered parameters

Movie(string paramName, string paramRating, int paramGross){

name = paramName;

rating = paramRating;

gross = paramGross;

}

};

**Main.cpp (Used to test my class)**

#include <iostream>

#include <string>

#include "class.h"

using namespace std;

int main(){

Movie myMovie = Movie("FIRST MOVIE", "PG-13", 1);

Movie oneMovie = Movie("FIRST MOVIE", "R", 2);

cout << myMovie.getName() << endl;

Movie combinedMovie = myMovie.mergeMovie(oneMovie);

cout << combinedMovie.getName() << endl;

cout << combinedMovie.getRating() << endl;

cout << combinedMovie.getGross() << endl;

};