Assignment 3

Deadline:	Friday 25th September 2015
Evaluation:	10 marks – which represents 4% of your final grade
Late Submission:	1 mark off per day late
Teams:	The assignment can be done individually or in teams of up to four people (one assignment per team submitted including all student names in the source code).
Purpose:	Practice with C++ classes, inheritance and polymorphism.

Problem to solve:
You have to write a program to be used by the Massey University Library. There are three types of items that can be borrowed from the library – Books, Journal articles and Movies. All of these items are a type of **Media** item. You will need to create a simple inheritance hierarchy for items that the library lends out. Create a **Media** base class with **Book**, **Movie** and **Journal** as derived classes as specified below.

Requirements:

The **Media** base class should contain the following **private** data:
id (integer), **title** (string), **year** (integer), **rating** (integer), **number of reviews** (integer)

Every item should initially be created without no reviews and no rating.

The **Media** class should provide the following functions:
Constructor
`rate(int rating)` a function that adds another rating to a Media item
`print(ostream &out)`
`printRating(ostream &out)`

The **Movie** derived class should contain the following data:
director (string)

It should also contain the following functions:
Constructor
`print(ostream &out)`

The **Book** derived class should contain the following data:
author (string)
number of pages (integer)

It should also contain the following functions:
Constructor
`print(ostream &out)`

The **Journal** class should contain the following data:
volume (integer)
number (integer)

It should also contain the following functions:
Constructor
`print(ostream &out)`

You should also write an output operator (`operator<<`) that will output a **Media** object. This should print out the appropriate information based on what class the object is an instance of (hint: use the `print` function).

You **must** follow the next three specifications in **each and every assignment for this course**

1. Place the following comments at the top of your program code and **provide the appropriate information**:

```
// Family Name, Given Name, Student ID, Assignment number, 159.234
/* explain what the program is doing . . . */
```

2. Create the function *displayInfo* as shown below and **provide the appropriate information**:

```
void displayInfo() {
   cout << "----------------------------------------" << endl;
   cout << "Assignment 3 Semester 2 2015"            << endl;
   cout << " Submitted by: Mouse, Mickey, 12345678"  << endl;
   cout << "          and  Duck,  Donald, 98765432"  << endl;
   cout << "----------------------------------------" << endl;
   cout << endl;
}
```

The *displayInfo* should be the first thing that you display on the screen. If I supply the *main()* then you are responsible only for the implementation of the function *displayInfo* and I will call it in my *main()*.

3. **DO NOT** use any function to clean the screen at any stage of a program, for any assignment in this course.

Hand-in: Submit **a3.cpp** electronically on the course stream site.

Marks will be allocated for: correctness, completeness, no code duplication, use of C++ tools, documentation, clear on screen output display**…**

If you have any questions about this assignment, please ask the lecturer.

The Program:

```
int main() {
  Media *ptr[10];
  ptr[0] = new Movie(352, "Remember The Alamo", 1945, "George Smith");
  ptr[1] = new Movie(831, "High School Blues", 1984);
  ptr[2] = new Movie(194, "Going for the Touchdown", 1984, "Frank Madden");
  ptr[3] = new Movie(576, "Martian Hairdresser", 1992, "Debbie Gold");
  ptr[4] = new Book(608, "How to Make Money",1987, "Phil Barton", 324);
  ptr[5] = new Book(442, "Garden Projects At Home", 1998, "Mary Freeman", 164);
  ptr[6] = new Book(185, "The Haunted House Mystery", 1996, "Bert Morgan", 53);
  ptr[7] = new Journal(294, "ACM", 2009, 6, 8);
  ptr[8] = new Journal(521, "J of Logic", 2008, 23, 14);
  ptr[9] = new Journal(630, "J of AI", 2009, 35, 11);

  cout << "Printing 10 items..." << endl << endl;

  for (int i = 0; i < 10; ++i) {
    cout << *ptr[i] << endl;
  }
  ptr[0]->rate(3);
  ptr[0]->rate(1);

  ptr[2]->rate(4);
  ptr[2]->rate(2);

  ptr[3]->rate(1);
  ptr[3]->rate(1);

  cout << endl << "Printing again..." << endl << endl;

  for (int i = 0; i < 10; ++i) {
    cout << *ptr[i] << endl;
    delete ptr[i];
  }
}
```

Should produce the output:

```
Printing 10 items...

Movie: 352, Remember The Alamo, (1945), Directed by George Smith (no rating)
Movie: 831, High School Blues, (1984), Directed by unknown (no rating)
Movie: 194, Going for the Touchdown, (1984), Directed by Frank Madden (no rating)
Movie: 576, Martian Hairdresser, (1992), Directed by Debbie Gold (no rating)
Book: 608, How to Make Money, (1987), By Phil Barton, 324 pages (no rating)
Book: 442, Garden Projects At Home, (1998), By Mary Freeman, 164 pages (no rating)
Book: 185, The Haunted House Mystery, (1996), By Bert Morgan, 53 pages (no rating)
Journal: 294, ACM, (2009), Volume: 6, Number: 8 (no rating)
Journal: 521, J of Logic, (2008), Volume: 23, Number: 14 (no rating)
Journal: 630, J of AI, (2009), Volume: 35, Number: 11 (no rating)

Printing again...

Movie: 352, Remember The Alamo, (1945), Directed by George Smith (**)
Movie: 831, High School Blues, (1984), Directed by unknown (no rating)
Movie: 194, Going for the Touchdown, (1984), Directed by Frank Madden (***)
Movie: 576, Martian Hairdresser, (1992), Directed by Debbie Gold (*)
Book: 608, How to Make Money, (1987), By Phil Barton, 324 pages (no rating)
Book: 442, Garden Projects At Home, (1998), By Mary Freeman, 164 pages (no rating)
Book: 185, The Haunted House Mystery, (1996), By Bert Morgan, 53 pages (no rating)
Journal: 294, ACM, (2009), Volume: 6, Number: 8 (no rating)
Journal: 521, J of Logic, (2008), Volume: 23, Number: 14 (no rating)
Journal: 630, J of AI, (2009), Volume: 35, Number: 11 (no rating)
```