Write a program that uses the class `SalariedEmployee` from lecture 20 (slides 17 -19). Your program is to define a class called `Administrator`, which is to be derived from the class `SalariedEmployee`.

Your derived class should consist of the following member variables:

- A member variable of type `string` that contains the administrator's title (such as Director or Vice President).
- A member variable of type `string` that contains the division of administration (such as Research, Accounting, or Personnel).
- Two options to choose from for a member variable called `salary`. 1) You can change `private` in the base class to `protected`. 2) You can define another private member variable called `salary` in the derived class.

Also include the following member functions:

- A member function called `set_division`, which changes the division of the admin.
- A member function called `input_data` for reading in an administrator's data from the keyboard.
- A member function called `print`, which outputs the object's data to the screen.
- An *overloading* of the member function `print_check()` with appropriate notations on the check.

Implement and test this hierarchy that is now created (`Employee`, `Salaried Employee`, and `Administrator`). Test all member functions. A user interface with a **menu** would be a nice touch for your test program.

Note: Generate the classes in their respective files. Use namespaces with your name. There will be 7 files (3 interfaces, 3 implementations, 1 application file). Put them in a zip file and submit on to Canvas. The first two class definitions are already given to you. Your responsibility is to create the Admin class and to link them.

Have fun!