Dynamic Arrays, Pointers, and Operator Overloading

Summary: You will write a program to represent an Address Book with different functionality.

Functional Requirements:

1. Each entry in the Address Book has a **full name**, **email**, and **telephone number**.

2. The Address Book must be implemented as a dynamic array. The initialize capacity of the list is 1. Whenever the list needs to be resized, simply double its previous capacity. **You cannot use vectors.**

3. The functionality you need to support is as follows:

a) An **add** function to add an entry into the Address Book.

b) Override the **[] operator** to access the ith element of the list. If the index is out of bounds, return null.

c) Override the **<< operator** to print the list. Example of acceptable output:

```
------Address Book-------

Size: 3

John Doe

123-456-7890

john@doe.com

Jane Doe

451-231-2312

jane@doe.com

Patrick Jones

234-231-2312

patrick@gmail.com

---------------------------
```

d) Override the **+ operator** to return a new Address Book that has all the items of both Address Books

Suppose you had two address books, **a** and **b**, then **a + b**, would return a new Address Book that contains all the entries in **a** followed by all the entries in **b**. The entries ARE COPIES. This means that if I were to change an entry in the original address book **a**, then the corresponding entry in **a + b** will not be updated.

e) Override the *** operator** to return a new Address Book that has all the items of the Address Book multiple times.

Suppose you had an address book, **addrBook**, with 2 entries in it. Then, **addrBook * 3**, would return a new Address Book with 6 entries in it. You would make copies of the items in addrBook and do this process 3 times.