

Chris CullingLab report for **assignment 3**

8 Nov 2024

Exercise 1

int main()

```
PersonRegister magnusArchives(maxSize);
ReadRegister(magnusArchives, "PersonExempel.txt");
magnusArchives.Print();

AddPersonTest(magnusArchives);

RemovePersonTest(magnusArchives);

SearchByNameTest(magnusArchives);

SearchByAnyTest(magnusArchives);

EmptyRegisterTest(magnusArchives);
```

Outputs ☒

// Read from file and then print

```
-----
Arline Ansel, Jolene road 12, 000 00 PLEASE
Tess Addison, Englandismycity 1, 018 12 FACTS
Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Flossie Firmin, Fortenite 5, 999 99 GRIND
Oz Janson, Wizard Tower 4, 012 35 PLACE
-----
```

// Add Person Test

```
Attempting to add new person to register: YOUR NEW ROOMMATE, Your apartment
-----
Arline Ansel, Jolene road 12, 000 00 PLEASE
Tess Addison, Englandismycity 1, 018 12 FACTS
Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Flossie Firmin, Fortenite 5, 999 99 GRIND
```

```
Oz Janson, Wizard Tower 4, 012 35 PLACE
YOUR NEW ROOMMATE, Your apartment
-----
```

// Remove Person Test

```
Found and removed Arline Ansel, Jolene road 12, 000 00 PLEASE
-----
Tess Addison, Englandismycity 1, 018 12 FACTS
Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Flossie Firmin, Fortenite 5, 999 99 GRIND
Oz Janson, Wizard Tower 4, 012 35 PLACE
YOUR NEW ROOMMATE, Your apartment
-----
```

// Search-By-Name Test

1. search for non-existent
2. add new person via `AddToRegister(std::string& name, std::string& address)`
3. search for new person by name

```
Person Chell not found.
-----
Tess Addison, Englandismycity 1, 018 12 FACTS
Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Flossie Firmin, Fortenite 5, 999 99 GRIND
Oz Janson, Wizard Tower 4, 012 35 PLACE
YOUR NEW ROOMMATE, Your apartment
Chell, Aperture
-----
Person Chell found.
```

// Empty Register Test

The absence of persons between the divider lines in this output is the `Print()` of the now empty person register.

```
Emptying register.
Found and removed Tess Addison, Englandismycity 1, 018 12 FACTS
Found and removed Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE
Found and removed Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Found and removed Flossie Firmin, Fortenite 5, 999 99 GRIND
Found and removed Oz Janson, Wizard Tower 4, 012 35 PLACE
Found and removed YOUR NEW ROOMMATE, Your apartment
Found and removed Chell, Aperture
```


Exercise 2

// Freesearch Test (Search-By-Any)

To find matches via freesearch, I used `std::string::find()` on data strings (a `person`'s name and address combined in a single string) coupled with the search term.

This is how I tested it:

```
void SearchByAnyTest(PersonRegister &personRegister)
{
    string searchTerm = "e1";
    cout << "Freesearch results by search term " << searchTerm << ": " << endl;
    Person* searchResult1 = personRegister.SearchByAny(searchTerm, nullptr);
    Person* searchResult2 = personRegister.SearchByAny(searchTerm, searchResult1);
    Person* searchResult3 = personRegister.SearchByAny(searchTerm, searchResult2);
    if(searchResult1) searchResult1->Print();
    if(searchResult2) searchResult2->Print();
    if(searchResult3) searchResult3->Print();
}
```

Output ☒

```
Freeseach results by search term "e1":
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE
Chell, Aperture
```

Exercise 3

Constructor

```
PersonWithPhoneNumber::PersonWithPhoneNumber(string name, string address, string
phoneNumber) : Person{name, address}, phoneNumber{phoneNumber} {}
```

Print PersonWithPhoneNumber

```
Person* user = new PersonWithPhoneNumber("name", "address", "phone number");
user->Print();
```

Output ☒

```
name, address, phone number
```

Add to person register and print register

```
personRegister.AddToRegister(user);  
personRegister.Print();
```

Output ☒

```
-----  
Tess Addison, Englandismycity 1, 018 12 FACTS  
Kathlyn Simons, Simonsaid Lane 2, 012 34 PLACE  
Kelly Hallman, Ijustlivehere Street 5, 752 91 SOMEPLACE  
Flossie Firmin, Fortenite 5, 999 99 GRIND  
Oz Janson, Wizard Tower 4, 012 35 PLACE  
YOUR NEW ROOMMATE, Your apartment  
Chell, Aperture  
name, address  
-----
```

In `PersonRegister::Print()`, we're iterating through *pointers* to `Person` objects (not `PersonWithPhoneNumber` objects), and so it calls only `Person::Print()`!