

# frankSJSU DataStructure

 Search this site

## Frank's Home Page

## CMPE126 home

## Greensheet

## Frank's Notes

operator overloading  
storing objects  
pointer & deep copy  
array of objects  
linked list  
recursion  
stack  
queue  
search by hashing  
heap

## Frank's Slides

## Frank's Code

## Programming Exam

PE #3 F16  
PE #1 guide F15  
StudentGrade DB S14  
prg exam #4 S14

## Midterm Exams

midterm 1 S18  
midterm 2 F17  
midterm 1 F17  
midterm 2 S17  
midterm1 S17  
midterm2 F16  
midterm1 F16  
midterm S16  
midterm F15  
midterm S15  
bonus exam S15  
midterm F14  
midterm S14

## Final Exams

Final S17  
Final S16  
Final F15

[Labs and Homeworks](#) >

## Lab 1 classes

### Objectives:

1. Refresh and reinforce C++ object oriented design and implementations.
2. It is very essential that you are good at operator overloading ( ==, !=, <, >, <=, >= ). Why? Because it makes your life easier when there is any database like applications.
3. Get yourself familiar with the file processing (for lab 2). Why? Because in real life, data are stored in files.

## Lab 1: Refresh IDE and C++

### Overview

Use Eclipse to create a clockType with hr, min, sec as private members. You shall have 3 files: clock.h clock.cpp and lab1main.cpp

lab1main.cpp shall support the following statements:

```
clockType c1(15, 45, 30), c2(3, 20); // hour, min, sec
cout << c1; // add whatever to beautify it
cout << c2;
cout << c1+c2;
c2 = c1+c1;
cout << c2;
```

### Input Format:

none

### Output Format:

Enhance the sample main and display the output accordingly.

### Note:

1. You do need a constructor (why?). Under what usage scenario, you don't need any.
2. But, do you really need two constructors? Can you do it with one?
3. What's the difference between supporting `cout << c1 << c2;` versus two separate `cout`?
4. Do we need assignment operator overloading?
5. There are many sample code out there. Try to get "minimum" to support your user functionalities. By doing so, you will get better understanding of object programming.

### Homework:

Read in a file (complex.txt) which has a number of complex numbers in the form of  $a+bi$  (e.g.  $3+5i$   $2-3i$  etc).

1. **Create** a complex class to store data
2. **Input** each line in complex.txt(one complex object per line)
3. **Output** all complex objects to an output file named "complexObj.txt"

[Final S15](#)[Final S14](#)**Labs and Homeworks**[Misc Lab FYI](#)[Lab 0 C++](#)**[Lab 1 classes](#)**[Lab 2 object array](#)[Lab 3 Linked List](#)[Lab 4 Doubly Linked List](#)[Lab 5 Recursion](#)[Lab 6 Stack](#)[Lab 6+ math expression](#)[Lab 7 Simulation](#)[Lab 7a Palindrome](#)[Lab 8 search](#)[Lab 9 hashing](#)[Lab 10 sort](#)

4. **Compute** sum of all complex objects by overloading + operator and save value into result object

5. **Append** the result object to the end of "complexObj.txt"

Turn in a .zip file containing:

- main.cpp
- complex.cpp
- complex.h
- output file named "complexObj.txt"

Files should be in a runnable state.

Note: If the complex.txt file is too complex for you, you can change it a bit (e.g. take out the dummy line) to get your program going.



complex.txt (0k)

Frank sjsu Lin, Jan 24, 2...

v.3



## Comments

You do not have permission to add comments.

[Sign in](#) | [Recent Site Activity](#) | [Report Abuse](#) | [Print Page](#) | Powered By [Google Sites](#)