

Software Architecture

Hand-in Assignment

Week 2

Instructions:

- Submit your solutions through GitHub Classroom and **remember** to submit a link to your repository in BrightSpace. Otherwise, I will have a hard time distinguishing between the different groups.
 - When submitting, you should push your solutions to the existing repository. Create a folder called eg. "Solution" with your solution file(s) inside and push to the repository.
 - You **must** add comments in your programs to explain your code. Any solutions submitted without any comments will not be graded.
-

GitHub Classroom invitation link for this assignment: [GitHub](#)

Exercises

You are given two implementations of a ShoeShop data structure. One direct implementation (`../ShopOrg`) and one using an abstract class to achieve polymorphism (`../ShopAbs`). The class has a collection of function members to modify and query the state of a shop. You are not required to understand what the code is about. Only the structuring of the code and its representation in C are important. (So now you got to recall the C you have learned from the last semester.)

(0) **Read** the instructions above please :)

- (1) Implement the code of ShopOrg in C using structs instead of classes. This code directly links the functions to the ShopOrg class. No function pointers are required (but could be used, of course).
- (2) Implement the code of ShopAbs in C emulating the polymorphism provided by means of the abstract class Shop. In this implementation function pointers are required. They need to be implemented as members of the struct Shop. The implementation of ShoeShop must declare the function pointers in exactly the same order. A type cast to `'(Shop*)'` needs to be used to use a ShoeShop via a Shop pointer:

```
1 ShoeShop* shoeshop = malloc(sizeof(ShoeShop));  
2 // do something with shoeshop  
3 Shop* shop = (Shop*) shoeshop;  
4 // do something with shop
```