Midterm 2

1) a. List 4 key activities for object design

1) Identification of existing solutions (reusability)
   - Use of inheritance
   - Off-the-shelf components and additional solution objects
   - Design Patterns

2) Interface Specification
   - Describes precisely each class interface

3) Object model restructuring
   - Transforms the object design model to improve its understandability and extensibility

4) Object model optimization
   - Transforms the object design model to address performance criteria such as response time or memory utilization

Marisa Loraas

SWE 2nd Midterm Exam

b) two techniques for reusing functionality:

1) Inheritance

-Advantages: Straight forward, supported by many programming languages, easy to implement new functionality

-disadvantages: Inheritance exposes a subclass to the details of its parents. Any change in parent class implementation forces the subclass to change

2) Delegation

-Advantage: any object can be replaced at run-time by another one (as long as they are the same type)

-D.s advantage: Harder to understand. Efficiency

c) The "Programming to an interface, not an implementation" is often seen in the Strategy Pattern, and its used in order to encapsulate behaviors of a system into an interface, that can then be implemented by inherited classes. In the strategy Pattern this allows us to create families of related algorithms and more choice to implementation →

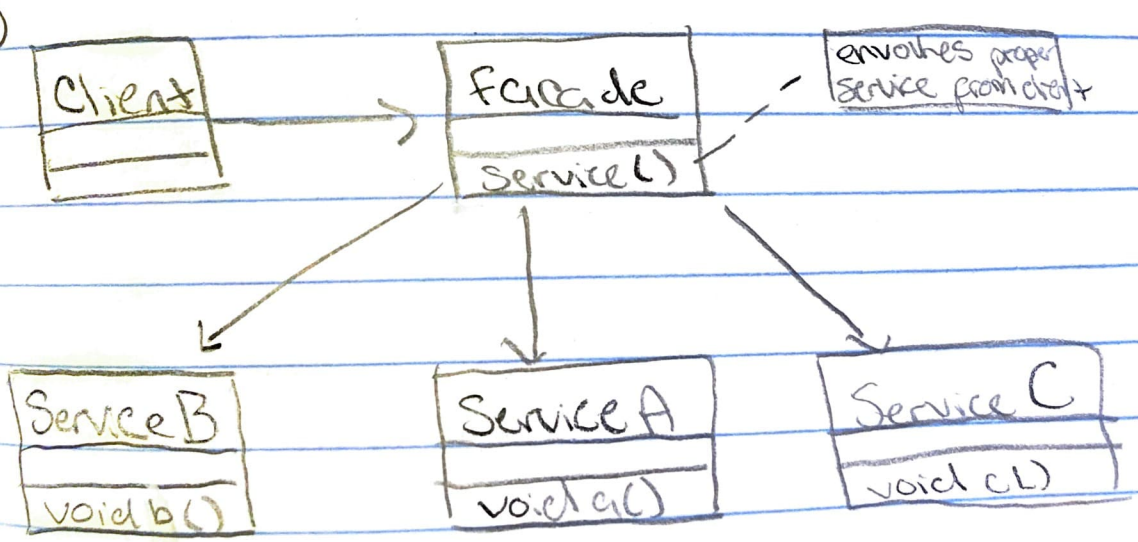
A good example of this would be for say an animal interface, and all the subclasses are specific animals that inherit from this class.

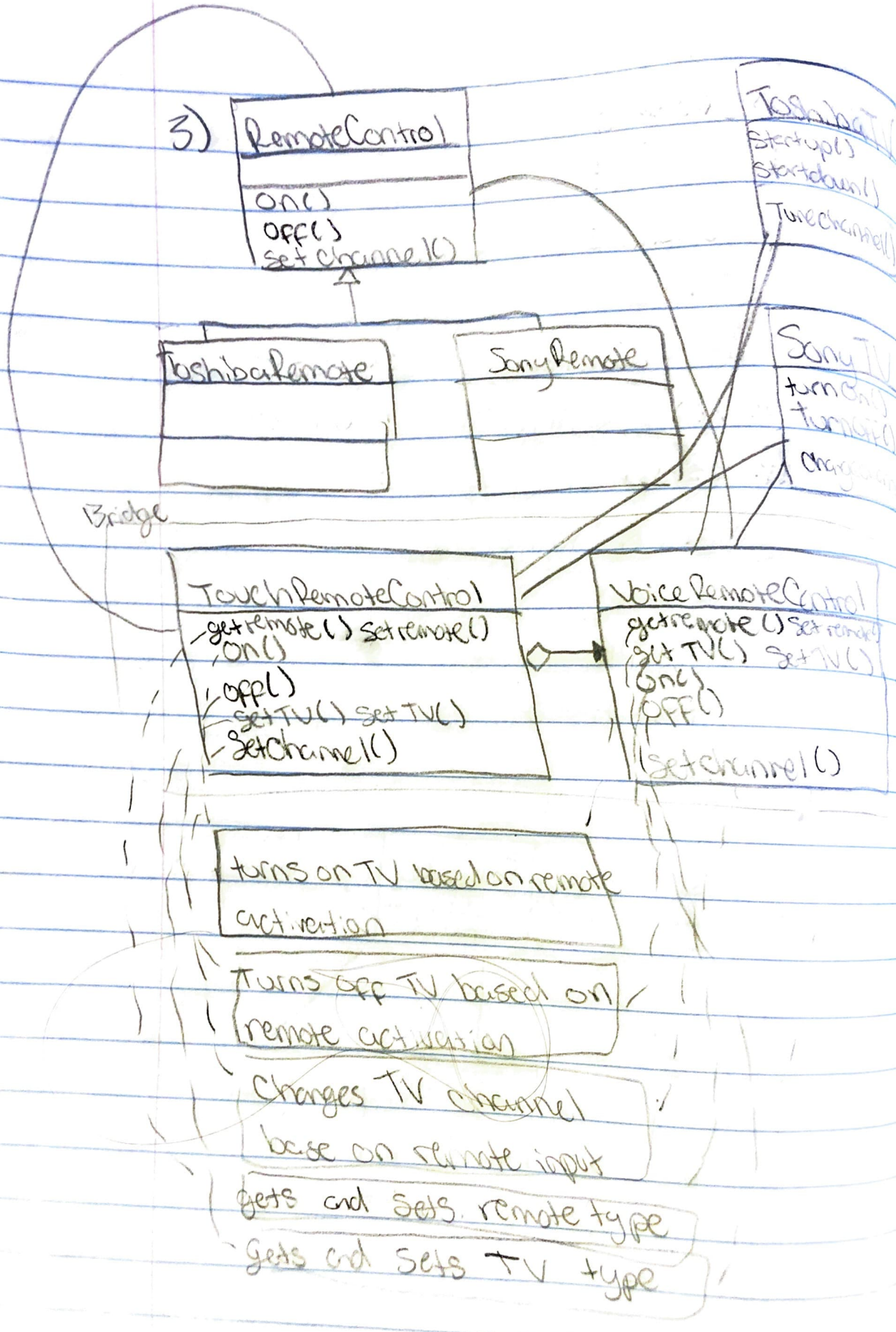
2) a) Design Patterns describe a problem which occurs over and over again in our environment. Then it describes the core solution to that problem, in such a way that you can reuse that solution over and over again.

four essential elements:

- Pattern Name
- Problem
- Solution
- Consequences

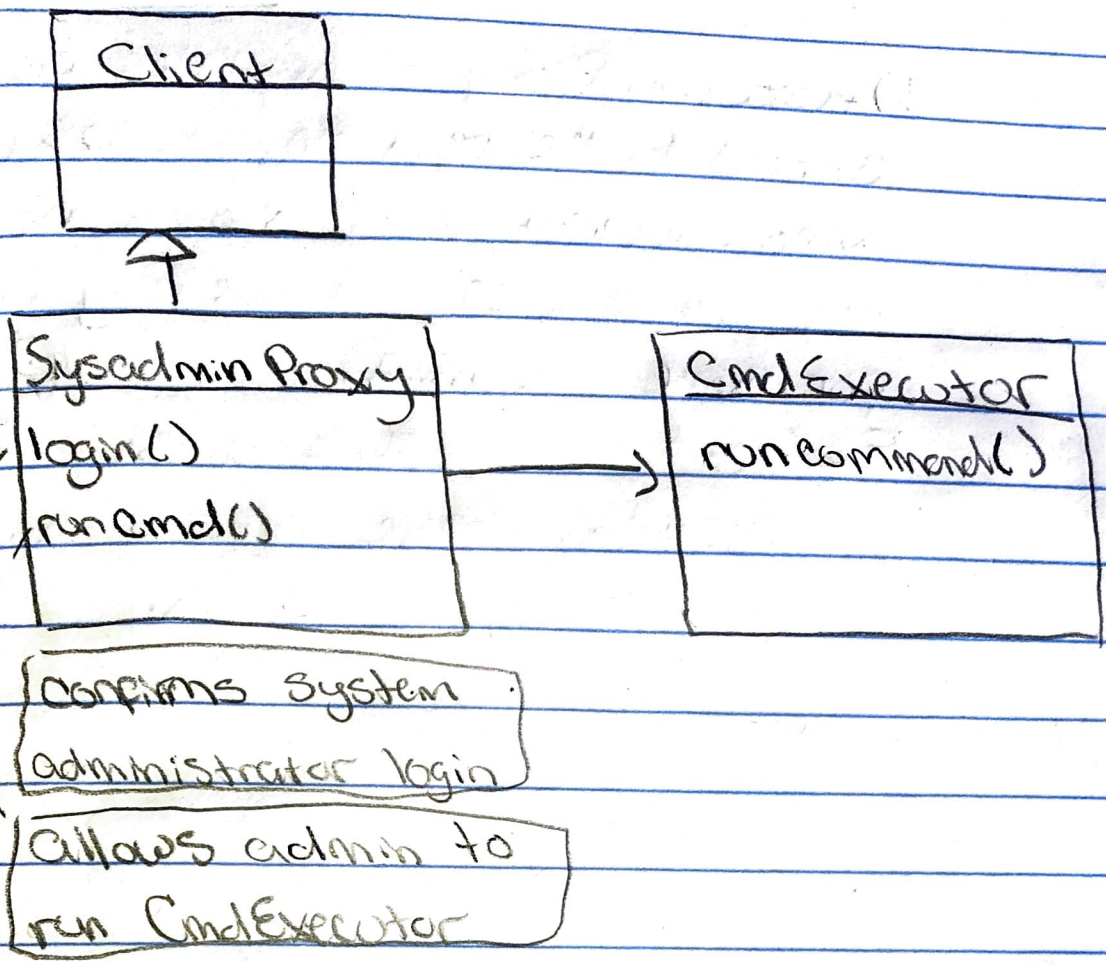
b)




3)
RemoteControl
on()
OFF()
setChannel()
ToshibaRemote
SonyRemote
Bridge
TouchRemoteControl
getRemote() setRemote()
On()
Off()
setTV() setTV()
SetChannel()
VoiceRemoteControl
getRemote() setRemote()
getTV() setTV()
On()
OFF()
setChannel()
turns on TV based on remote activation
Turns off TV based on remote activation
Changes TV channel base on remote input
gets and sets remote type
gets and sets TV type

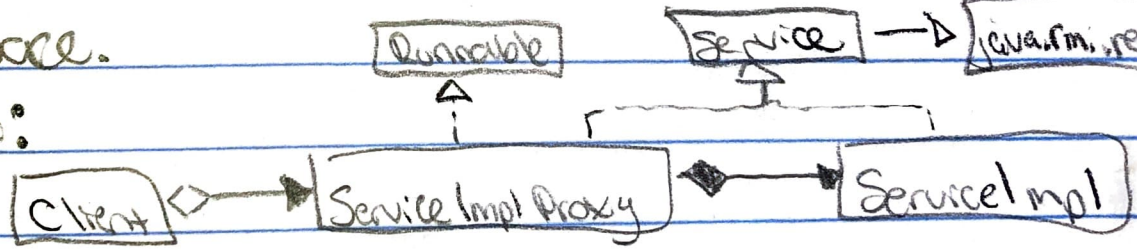
a)

4)



b) 1) Remote Proxy → Local representative for an object in a different address space.

Ex:



2) Virtual Proxy → Object is too expensive to create or too expensive to download.

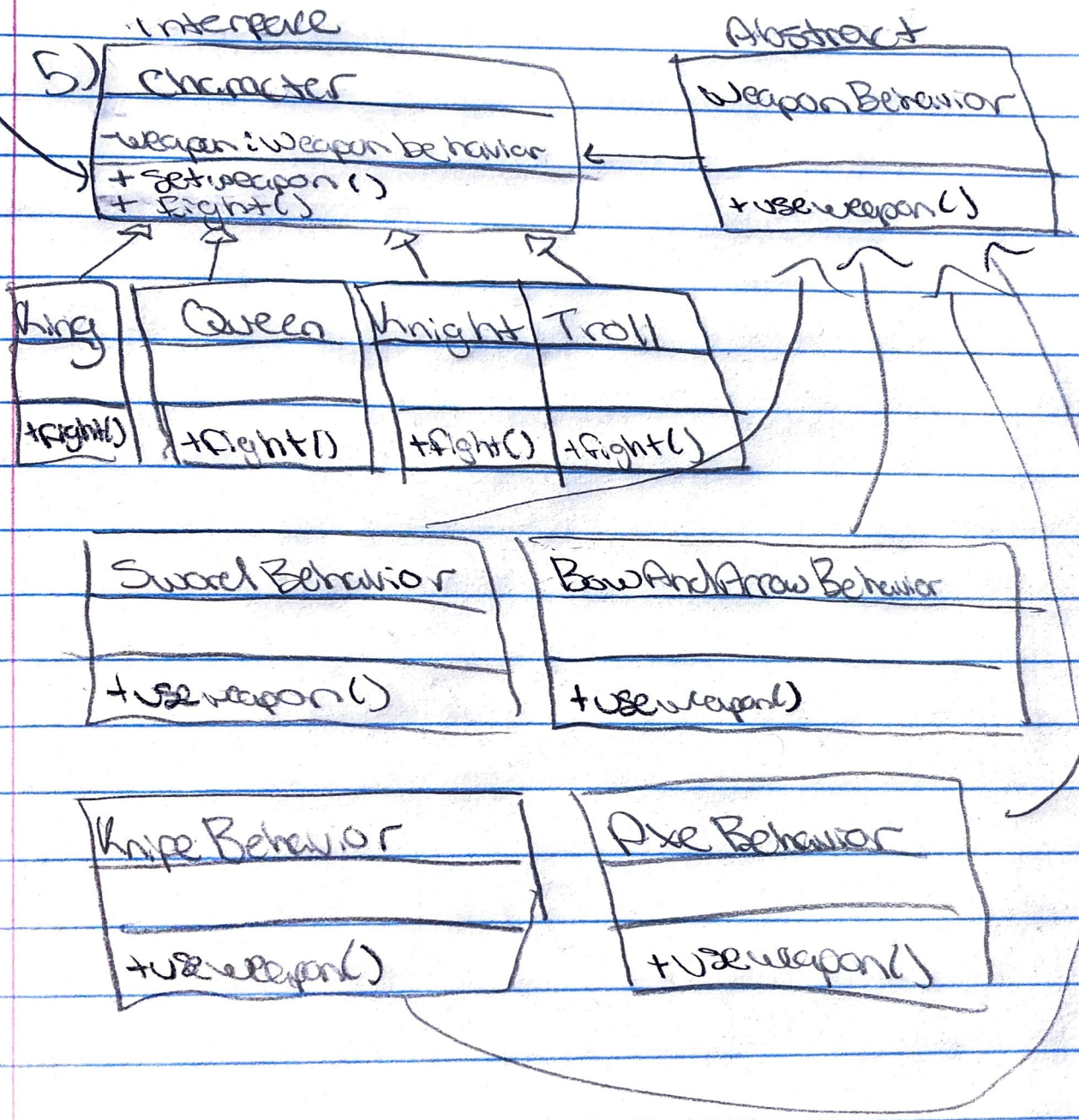
Proxy is a stand-in.

Ex: The ImageProxy example with Graphic class in the notes (slide 17, lec 14)

3) Protection Proxy → provides access control to the real object. Useful when different objects should have different access and viewing rights for some document.
Ex: part a of this question: giving sysadmin more access control than a regular client.

part b

5)


Interface
Character
-weapon:WeaponBehavior
+setWeapon()
+fight()
Abstract
WeaponBehavior
+useWeapon()
King
+fight()
Queen
+fight()
Knight
+fight()
Troll
+fight()
SwordBehavior
+useWeapon()
BowAndArrowBehavior
+useWeapon()
KnifeBehavior
+useWeapon()
AxeBehavior
+useWeapon()