

Lab 5 - Solution

Part I – Stack class:

Method	Pre-Condition	Post-Condition
peek()	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns the top of the stack• If empty, returns a null• Stack is unchanged
isEmpty()	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns true if the stack has no element, otherwise, it returns false• Stack is unchanged
isFull()	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns true if the stack has 1 or more elements, otherwise it returns false• Stack is unchanged

Part II – Set class:

Method	Pre-Condition	Post-Condition
insert(x)	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• x is now in the set• there are no duplicate copies of x in the set• the set's size increases by 1 if x is not in the set already
remove(x)	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns true if the set had x• If set has x, it no longer has x, and the size decrements by 1• Returns false if the set did not have x• If set is empty, it will always return false
isIn(x)	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns true if x exists in the set• Returns false if x does not exist in the set• Set does not change

Part III – Account class:

Method	Pre-Condition	Post-Condition
getBalance()	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns the balance of the account• Account balance is unchanged• The return value is of type real• Value is of 2 decimal precision
setBalance(real b)	<ul style="list-style-type: none">• b must be a real positive number	<ul style="list-style-type: none">• The account balance changes to value b
getCustomerName()	<ul style="list-style-type: none">• None	<ul style="list-style-type: none">• Returns the name of the owner of the account• the return is of String type

Part III – AccountOperations class:

Method	Pre-Condition	Post-Condition
withdraw(Account acc, Real am)	<ul style="list-style-type: none">• am must be a positive real number• acc exists and is an account• acc has a balance of am or more	<ul style="list-style-type: none">• acc decreases by am
balance (Account acc)	<ul style="list-style-type: none">• acc exists and is an account	<ul style="list-style-type: none">• The am in acc is returned• The am in acc is unchanged
transfer (Account from, Account to, Real am)	<ul style="list-style-type: none">• from is an account and exists• to is an account and exists• am is a positive real number• from account have same owner as to account• from balance has at least am or more	<ul style="list-style-type: none">• from balance decreases by am• to balance increases by am
deposit(Account acc, Real am)	<ul style="list-style-type: none">• acc exists and is an account• am must be a positive real number	<ul style="list-style-type: none">• acc balance increases by am

Part IV – MaxIntSet vs. IntSet

MaxIntSet would be a subtype of IntSet as it only builds onto the post conditions of IntSet and does not change any of the restrictions on the methods

Part V – LinkedList vs. Set

Set (S) is not a subtype of LinkedList (L) as the precondition of the set is that the element being added is not in there already. A program (P) cannot add its element to the set if it is using Set as set will not accept it. P will need to know if it is using S or L

Part V – Circle vs. Ellipse

A circle is not a subtype of ellipse since P will need to know if it is working with Circle (i.e. needs to ensure x and y are the same) in comparison if it is working with an ellipse (i.e. does not need to know if it x and y are the same)

Part VI – Stack vs. Vector

Not a subtype as we increase the preconditions in the stack

Part VII – LibraryEmployee vs. Clerk

Clerk is not a subtype of LibraryEmployee as it cannot do everything that an Employee would do and from a program P perspective, it needs to know if it is working with a LibraryEmployee or a Clerk to ensure that it only has the methods it has

Part VIII – Person and Employee

No it is not a subtype of a person as the post condition of the employee would be returning a the “:” instead of just the first name and the last name

Part IX – Different Abstract Situations

1.
 - a. Precondition Holds: true
 - b. Postcondition Holds: true
 - c. LSP holds: true
2.
 - a. Precondition Holds: true
 - b. Postcondition Holds: true
 - c. LSP holds: true
3.
 - a. Precondition Holds: false
 - b. Postcondition Holds: true
 - c. LSP holds: false
4.
 - a. Precondition Holds: true
 - b. Postcondition Holds: true

- c. LSP holds: true
- 5.
- a. Precondition Holds: false
 - b. Postcondition Holds: true
 - c. LSP holds: false

Preconditions: are broader for the child
Postcondition: are restricted for the child