Team 4: Authority Publishing Platform

- Seeding Defect Log -

Description	Output	Output	Suggested Corrections		
	Before	After		W or B	
Email.java checks for length of email, but does not verify that there is "@" or ".something" or check for garbage entries so someone could enter an invalid email address.	Email class checks for at least an "@"	Emails do not require an @ sign	Scan for @ sign in String	В	1
Citation class description and reason can be blank: non-blank tested using >=0.	Citations desc shouldn't be blank	Real Bug: Citations Desc. can be blank	Check if size is greater than zero, if not throw a polite error.	В	2
In Email.java isValidAddress() should check if address length is not zero.	Can't have zero length emails	Can have zero length emails	Check email length, do not allow blank emails	В	1
In Email.java isValidAddress() the .indexof("@") doesn't catch when an address starts with @	Can't have emails like "@example. com"	Can have emails like "@example. com"	Check the first character of the String for an @ sign	В	2
Server cannot be ran in a place with daylight savings: once a year sessions will expire an hour early.		Real Bug: Sessions use Date() func and daylight savings	Use a UTC clock and move daylight savings to the display layer rather than the time delta layer.	W	3

ClaimToken and RecoveryToken need to have appropriate lifetimes	Should expire after certain amount of time	Real Bug: They don't expire	After a predetermined amount of time database entries are garbage collected	w	3
Citation doesn't check length of description for very large inputs	Citation rejects strings over 255	Citation cuts off strings silently over 255	Instead of trimming strings silently and loosing data, throw a polite error when strings are too large.	В	3
viewPortfolio() only shows every other Citation in the Portfolio	Every Citation is printed	Every other Citation is printed	Remove a call to iterator.next() in viewPortfolio() in Cli.java	В	1
For removeFromPortfolio if invalid Citation ID is entered for removal an IllegalArgumentException will be thrown	"Invalid Citation" printed	Exception is thrown	add a try-catch block to the method that catches an IllegalArgumentException	В	2
viewPortfolio() doesn't check for the case of no Citations in a portfolio, so a NoSuchElementException is thrown if this method is called for an empty portfolio	Should print correctly	Exception is thrown	add a check to see if the Citation is null	В	1
For viewPortfolio() if the email passed as an argument contains a 'k' or an 'm' even if that email does exist the function will falsely return that it does not.	Should print correctly	prints false	Remove case for these letters so strings containing them are treated as the rest.	В	3

single insertion	duplicate insertion	Remove the extra call to add the Citation	В	1
Should add citation without error	false print, no insertion	Remove case for these letters	В	1
insertion and print confirming	false print, insertion	Remove case in addToPortfolio	В	1
Should check citation	ignores session key, can throw null pointer exception	Insert code to check for valid session, like if (!res.success) return null;	В	2
Should throw excpetion when length too large	Silently ignores condition	throw IllegalArgumentException in Citation.java and catch it in Cli.java	В	3
Should throw excpetion when length too large	Silently ignores condition	throw IllegalArgumentException in Citation.java and catch it in Cli.java	В	3
	Should add citation without error insertion and print confirming Should check citation Should throw excpetion when length too large Should throw excpetion when	Should add citation without error insertion and print confirming Should check citation Should throw exception when length too large Should throw exception when length too large Should throw exception when length too large Silently ignores condition Silently ignores condition Silently ignores condition	Should add citation without error Insertion and print confirming Should check citation Should check citation Should throw exception when length too large Should throw exception Should throw exception Should throw exception Should throw exception Silently ignores throw in Citation. Java and catch it in	Should add citation without error Should add citation without error False print, no insertion Remove case for these letters Balance

If description of Citation contains a space it is replaced with all spaces	proper insertion	blank insertion	Remove case in addToPortfolio()	В	1
If resource of a Citation contains a space it is replaced with "Resource"	correct resource added for Citation	blank resource added	Change code in addToPortfolio to handle spaces the same as other characters	В	1
addToPortfolio does not check that a valid session was passed to it.	Should check citation	ignores session key, can throw null pointer exception	Add case to addToPortfolio to check for this, such as if (!res. success) return new Result (false, "Invalid session");	В	2
dateCreated() for Citation is an hour earlier than it should be	Should be correct date	Date is an hour earlier than should be	Fix time in dateCreated in Citation	В	2
Email class does not check the length of the input. Allows 1 character emails.	Emails such as '@' or blank emails are invalid.	Can show blank or 1 character emails as valid	Add a check to the Email class validateEmail() to ensure that its length is greater than 1	W	2
Email validation fails on an email that begins with the character 'a'.	Emails are validated correctly	Emails are seen as invalid if they begin with an 'a'	Replace 'b-z' with 'a-z' in the first regex group in the Email class, in EMAIL_PATTERN	В	3

Email validation fails on an email with a digit (0-9) directly after the '@' sign.	Emails are validated correctly	Emails are seen as invalid if they have a digit after the '@'	Add '0-9' to the EMAIL_PATTERN regex in the Email class, in the grouping right after the '@'	В	3
Email validation fails if the address contains "gmail".	Allows the address and validates correctly	Prints "No gmail! The NSA is watching!"	Remove the bad validation line in validateEmail()	В	2
Email validation fails if the address contains "bob".	Allows the address and validates correctly	Prints "Crazy bob rejects this email!" and returns as an invalid address	Remove the bad validation line in validateEmail()	В	3
addEmail() functionality allows any claim token to match with an email.	Only adds emails when given a valid claim token	Will add any email if given any claim token	Add check of the given ClaimToken against the ClaimToken of the given email.	В	1
addEmail() does not check the result of validateEmail() and will allow invalid emails to be added to a user.	Prints "invalid email, no ClaimToken"	Allows invalid emails to be added to a user	Check to see if the 'success' field of the Result returned from validateEmail() is 'true'	В	2
addEmail() does not remove claim token, allowing emails to be registered with multiple users	Emails are tied to a single account	Emails could be claimed after already being clamed by a user	Should remove database relationship to avoid error state	В	1

Password validation fails for the string 'password'.	Allows the password and validates correctly	Prints "No lazy passwords!" and returns as an invaild password	Remove the bad validation line.	В	2
Email.getAddress() replaces 'b' with 'X'	Returns email correctly	Returns an incorrect email string	Remove the replace() call in getAddress() that mangles the address String.	В	2
Email.getAddress() replaces '@' with "AT"	Returns email correctly	Returns an incorrect email string	Remove the replace() call in getAddress() that mangles the address String.	В	2
addEmail() does not validate that the session before operating. Bad sessions crash.	Prints "Invalid Session"	Crashes	Check the Result returned from validateSession()	В	3
deleteEmail() does not validate that the session before operating. Bad sessions crash.	Prints "Invalid Session"	Crashes	Check the Result returned from validateSession()	В	3
deleteEmail() will not delete emails containing 'uk'.	Prints "Removed Email"	Prints "UK emails are permanent!"	Remove the bad validation line.	В	1

deleteEmail() returns a 'False' Result even if it successfully removes the email.	Prints "Success: Email Removed"	Prints "Failure: Email Removed"	Change success argument of final Result returned by deleteEmail() to be 'true', not 'false'.	В	1
viewSubjectiveNetwork crashes when given invalid session	Print "invalid session"	Crash	Call invalidateSession()	w	3
viewSubjectiveNetwork doesn't go far enough	go to the correct depth	goes one level not far enough	Go to threshold, not threshold -1	b	2
viewSubjectiveNetwork() prints out all emails for each user traversed	Prints one email per user	Prints all emails per user	Add a break statement after the first email is printed	b	3
viewSubjectiveNetwork() doesn't check if subjects match	Prints the subjective network from a users perspective	Prints all subject trust networks	Check for edge subject is equal to the requested subject	b	1
viewSubjectiveNetwork() doesn' t print the trust edges from each email owned		Take away the break statement	Add break statement	b	2

viewSubjectiveNetwork() doesn' t print the user emails in an ordered manner	Should print in a "nearest" first list	Prints in a random order	Add to the queue first not last	b	3
viewSubjectiveNetwork() always returns false Result	Should return True result	Returns False Result	Switch result to True	b	3
viewSubjectiveNetwork() prints everything on the same line	Should print one email per line	Prints everything on the same line	Change System.out.print to System.out.println()	b	2
viewTrustNetwork() doesn't check if address is a valid address	Tell user the entered an invalid email	Lets user use bad emails	Use java regular expressions to check	b	3
viewTrustNetwork() doesn't check if there is a User object associated with the email	Could crash if it is an unregistered email	Crashes if unregistered email is passed	Make sure a valid User object is found.	b	2
viewTrustNetwork() starts from a single email and not all emails	Doesn't check the trust from each email	Only prints trust network of certain emails	Take away the break statement before the BFS in viewTrustNetwork()	b	1

viewTrustNetwork() doesn't print the subject on each line	Subject is on the same line as emails	Subject should be on its own line	Change System.out.print to System.out.println	b	2
viewTrustNetwork() prints all the addresses on the same line	Emails should be on their own line	Emails are on the same line	Change System.out.print to System.out.println	b	2
viewTrustNetwork doesn't check if an Email object has an email	Check if it's not null	Crash with bad input	Add check	b	3
viewTrustNetwork() doesn't check if a User has no Email	Skip it	Crash	Check if User is not null	b	2
viewTrustNetwork() prints all of the emails of each traversed User	Should only print one	Prints them all	Add a break after the first email is printed	b	2
viewTrustNetwork() always returns false	Should return true	Returns false result	Change false to true	b	2

viewTrustNetwork() always prints "subject" instead of subject	Should print actual subject	Always prints "subject"	Change to the subject variable	b	3
viewTrustNetwork() checks all edges, not just outoing FROM edges	Should only traverse FROM edges	Traverses all edges	Change the getRelationships() arguments to FROM	b	1
viewTrustNetwork() prints extended network of trust and not just immediate.	Only prints direct trust edges.	Prints extended network.	BFS should be limited to depth one.	b	1
viewSubjectiveNetwork() checks all edges, not just outgoing FROM edges	Should only traverse FROM edges	Traverses all edges	Change the getRelationships() arguments to FROM	b	1
If email passed to viewPortfolio contains "edu" a blank portfolio is returned	view portfolio	blank result printed	Remove case from viewPortfolio	В	1
If the session ID contains a number for removeFromPortfolio it will try to delete the citation twice.	single deletion	incorrect deletion	Remove case from removeFromPortfolio	В	1

if Citation description contains "the" for viewPortfolio it is deleted when it is viewed	view portfolio	incorrect removal from portfolio	Remove case in viewPortfolio	В	1
Never expire Session's , they are valid for as long as the program is running	Sessions expire in 1 hour	Sessions never expire	Check in the `isValid` method for session expiration	W	1
Never check if the auto generated Session ID is unique	Session ID's are unique	Session ID's have a chance to override each other	Instead of assuming that the 4 chcharacteression ID is unique, attempt to check for uniqueness before returning it or return an error.	В	1
If login can return success with non-unique session-token.		Real Bug: If table is dense may return used id.	Currently: tries 20 times, if collides every time just returns last seen. Fix: return failure on all collisions, possibly have variable token size.	W	3
Login: Passwords are trimmed to five characters in login.	Full password is compared.	Only first five characters are compared.	Remove substring from createFromLogin (similar to real issue in login system).	В	2
Login: Trimming password assumes at least 5 characters.	Password trimmed to five or fewer	If less than five, throws except.	Only trim password if length is greater than five.	В	1

Register: Spaces and some special characters are silently removed from password.	Password does not accept special characters.	Password silently removes special characters.	Rather than removing illegal characters return an error condition.	В	2
Unused e-mail validation for signup	Email is checked using validation function	Email addresses are not validated before being used	Make call to validateEmail function or do some type of regex email address validation	В	1
Returning claimtoken during signup doesn't check to see if it is already claimed, throwing an exception if it is	Checked Email for present claimtoken. If it didn't exist user informed of taken email.	Removed check for present claimtoken.	Add check to make sure Email object has a claimtoken before attempting to retrieve it	В	2
Passwords are checked for common substrings instead of equality	Password and Verification are checked for equality.	Password and Verification are checked for common substrings	Check equality of password and verification password	В	1
Passwords ignore case, treat all characters as lower-case	Checked passwords as they were	Checks passwords after converting them to lowercase	Check equality of passwords based on original values	w	1
When registering, password saved to user is converted to all lower-case characters.	User password was set from the original password parameter	password is converted to lower case for equality checks before setting it in User	Make password set as User's password equal to the original parameter if it and it's verification are equal	В	1

Claimtoken isn't cleared by itself after registering an email.	Email's clearClaimToker method is called after registering	Method call is removed and no longer clears claimtoken	Make call to clearClaimToken() on Email to remove token	W	2
Registering with an invalid claimtoken throws exception instead of returning user output on the situation	User is presented with "bad claimtoken" message	Illegal State Exception is thrown	Catch exception on non- existent Email when searching the database	В	3
Registering with an invalid claimtoken throws exception instead of returning user output on the situation	User is presented with "bad claimtoken" message	Illegal State Exception is thrown	Catch exception on non- existent Email when searching the database	В	3
Registering account with password containing "andrew" replaces occurrences with "mack"	No substring replacement occurs	"andrew" within passwords replaced with "mack"	remove string replacement	В	1
Registering account with password containing "jason" replaces occurrences with "salter"	No substring replacement occurs	"jason" within passwords replaced with "salter"	remove string replacement	В	1
Registering account with password containing "james" or "jim" replaces occurrences with "miller"	No substring replacement occurs	"james" and "jim" within passwords replaced with "miller"	remove string replacement	В	1

Registering account with password containing "nathan" or "nathaniel" replaces occurrences with "cherry"	No substring replacement occurs	"nathan" and "nathaniel" within passwords replaced with "cherry"	remove string replacement	В	1
Registering account with password containing "mitchell" or "mitch" replaces occurrences with "mounts"	No substring replacement occurs	"mitchell" and "mitch" within passwords replaced with "mounts"	remove string replacement	В	1
Registering account with password containing "christina" replaces occurrences with "atallah"	No substring replacement occurs	"christina" within passwords replaced with "atallah"	remove string replacement	В	1
Using password that is a palindrome causes an exception	palindrome check doesn't exist	palindrome passwords throw palindrome exception	Catch palindrome exception or allow password palindromes	В	3
Passwords longer than 12 characters are truncated	no maximum length of password in register	checks password for length > 12	Remove cap on password length or inform user after throwing error.	W	3
Using email that is a palindrome causes an exception	palindrome check doesn't exist	any email that is a palindrome causes an exception to be thrown	Catch palindrome exception or allow usernames to be palindromes	В	3

Logging out doesn't check for valid session id before attempting to remove from session table	if session can't be validated, return	no check, attempts to remove from session table every time whether or not the session id is valid	Add if check to return result before removing from session table if session id is invalid.	W	3
Logging in with Konami Code always as password always works.	No master password	Konami Code master password	Don't use master passwords, it's a bad idea.	W	1
Citation's description argument for `trust` call is being reversed	Not reversing string input	String input is reversed	Don't reverse input	В	2
Subjects passed to `trust` will have a literal "\0" added to them	C style string error not present	C style string encoding error is present	Don't make string copy functions, use the library	В	3
clearClaimToken doesn't actually clear ClaimToken	ClaimToken cleared from DB	ClaimToken not cleared	Add a line of code to clearClaimToken() in Email.java to remove it.	w	2
`logout` command doesn't remove Session from table	When logging out, remove session object from hashtable	Session is not removed from table	Remove session from table	В	1

		I .			_
If Citation description contains "of" it will get re-added to the portfolio when viewPortfolio() is called	view portfolio	add extra Citation for view portfolio if description contains "of"	Change viewPortfolio() to treat descriptions containing "of" the same as others	В	2
TrustEdge.toString() doesn't print matching brackets	Should be {}	Is {)	Change the print statement in the toString() function	В	3
If session_id contains a "Y" and addToPortfolio is called invalid session will be falsely printed	Add citation	incorrect print	Remove case from addToPotrfolio	В	2
Email addresses with case-insensitive equality will collide.	is distinct from e@example.	collides with e@example.com	remove toLowerCase() from email constructor.	В	2
All Subjects now end with " fizzbuzz"	Subjects could print fine	Subjects have a tacky "fizzbuzz" added to the end	Stop adding fizzbuzz to the subjects	В	1
If resource contains wikipedia or reddit resource is instead [Citation Needed]	Resource is what the user meant.	Resource is [Citation Needed]	Just keep resource as what user meant.	W	3

The Unlucky rows of viewTrustNetwork are not printed	All rows were printed, regardless of subjective luck	Unlucky #13 rows aren't printed, or every 13th row is skipped	Instead of skipping every 13th row, print it instead	В	1
The lucky rows of viewSubjectiveNetwork are not printed	All rows were printed, regardless of order	Lucky #7 rows are skipped, their secrets are lost to the aybis of time	Instead of skipping Lucky #7 rows, don't	В	1
If "ascii art" is passed to viewSubjectiveNetwork then bad results are printed	No bad input	Dancing ascii man is printed	Remove print statement with the dancing ascii man	w	3
When logging out if session contains a 7, logout appears to fail	Logging out any session id works fine	Logging out with a session id that contains a "7" will fail	Logout should not fail depending on id, remove code that fails	В	3
When calling viewSubjectiveNetwork with a subject of "app", program errors	Program can print a network based on "app" fine	Program refuses to print due to a bug	Remove bug code and allow for printing	w	3