# GIT Department of Computer Engineering

# CSE 222/505 - Spring 2023 Homework #4 Report

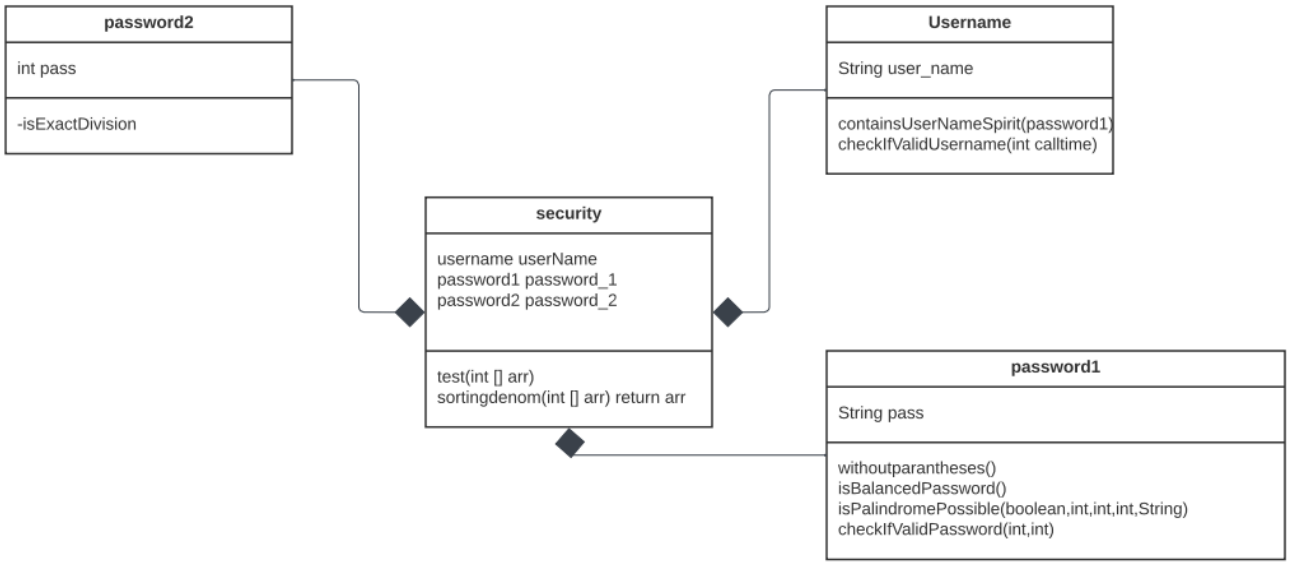
**Pelin Erdoğan**

**210104004266**

## 1. SYSTEM REQUIREMENTS

When the program executes even though every element can be created separately you can just create a security object. Security object has username,password1 and password2. And security class has a test function to test the program. Test function takes denominations array as a parameter and function should be sorted highest to lowest. That's why class also has a sort array function.

## 2.USE CASE AND CLASS DIAGRAMS




## 3. TİME COMPLEXİTY DİAGRAM

| | |
|---|---|
| **checkvalidusername** | **O(n)** |
| **checkvalidpassword** | **O(n)** |

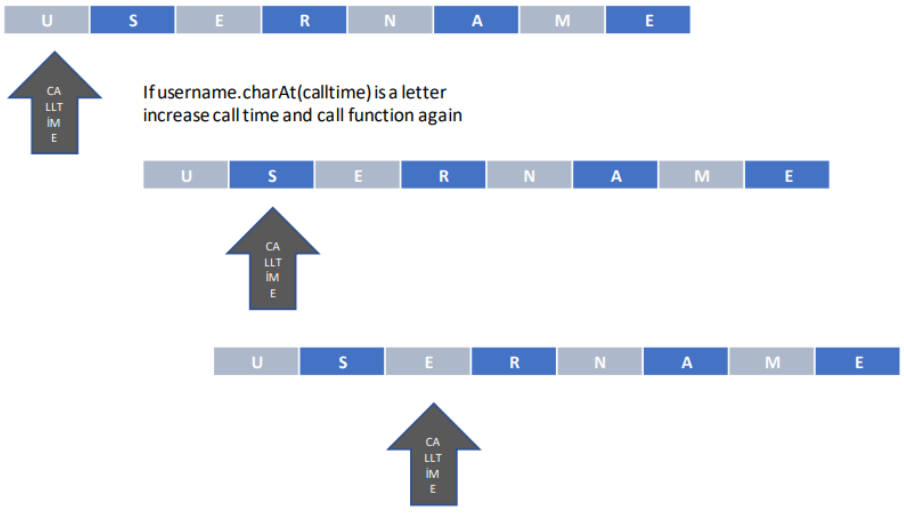
| | |
|---|---|
| **containsusernamespirit** | **O($n^2$)** |
| **isBalancedPassword** | **O(n)** |
| **isPalindromepossible** | **O($n^2$)** |
| **isExcactdivision** | **O(n)** |
| **Sortingdenom** | **O(n)** |
| **test** | **O(1)** |

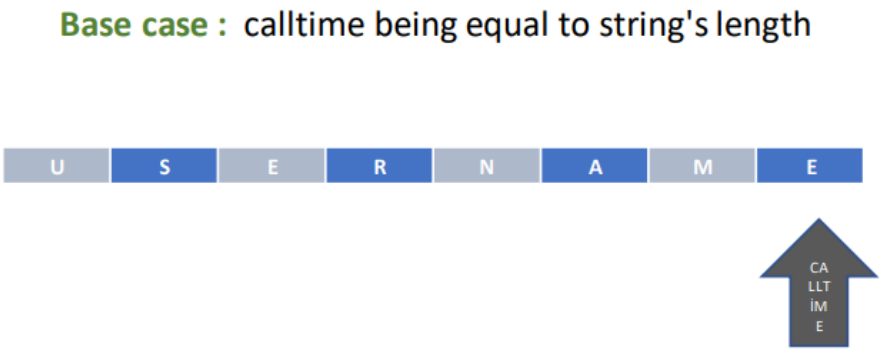
## 4. PROBLEM SOLUTION APPROACH

### Problem solution of checkvalidusername:

In this function I check if username length is 0 if so return false.

Then I check if calltime is smaller than length of username. If username length is equal to calltime function ends. If it is smaller function checks character in string using ascii values of letters. If character is a letter function calls itself with increasing calltime. If character is not a letter function returns false.



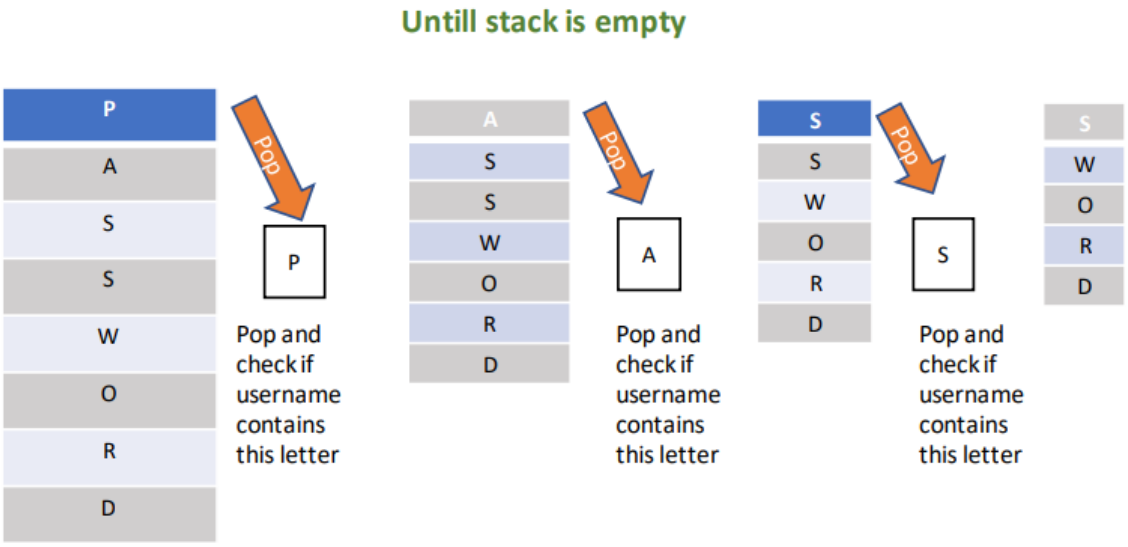



**Problem solution of checkvalidpassword:**

This function almost works exactly like checkvalidusername. But it returns false if password length is smaller than 8 and while it is calltime is searching through password it counts parentheses.

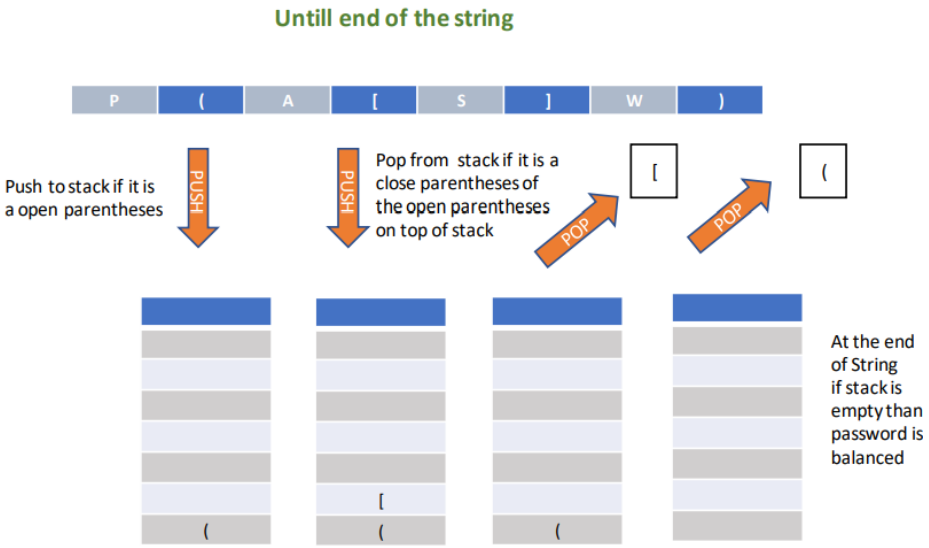
**Problem solution of containsusernamespirit:**

In this function characters of password are pushed to stack. After that every element is popped and checked if username contains them



**Problem solution of isBalancedPassword:**

In this function with a for loop I search through the password and If a character in String is an open parenthesis, I push it to stack. If a character is a close parenthesis, Function checks if that close parenthesis is matching with the open parenthesis on top of the stack. If it is matching, Function pops the open parenthesis in stack. If it is not matching it returns false. When the function reaches the end of the string if stack is empty, it returns true. Else it returns false.



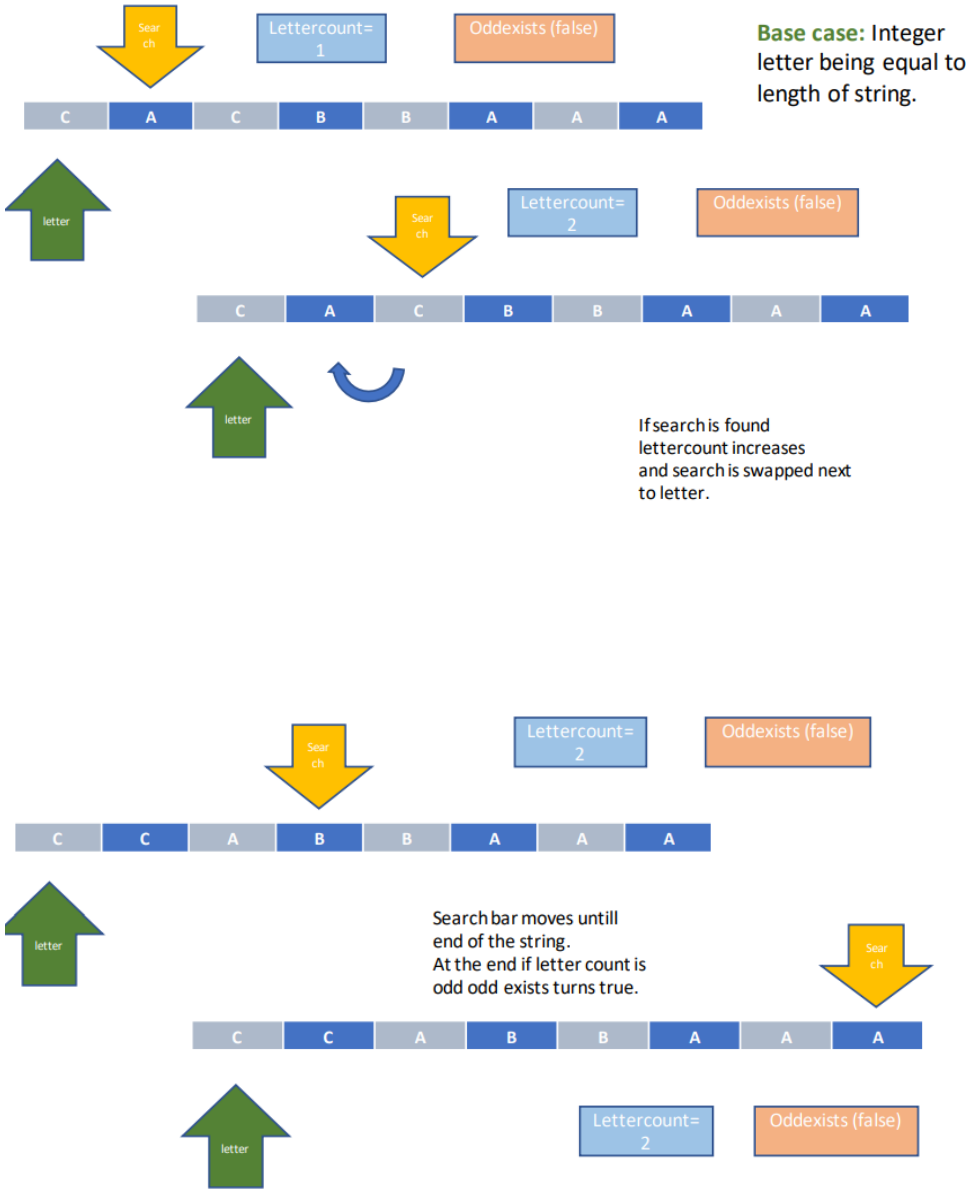

**Problem solution of isPalindromepossible:**

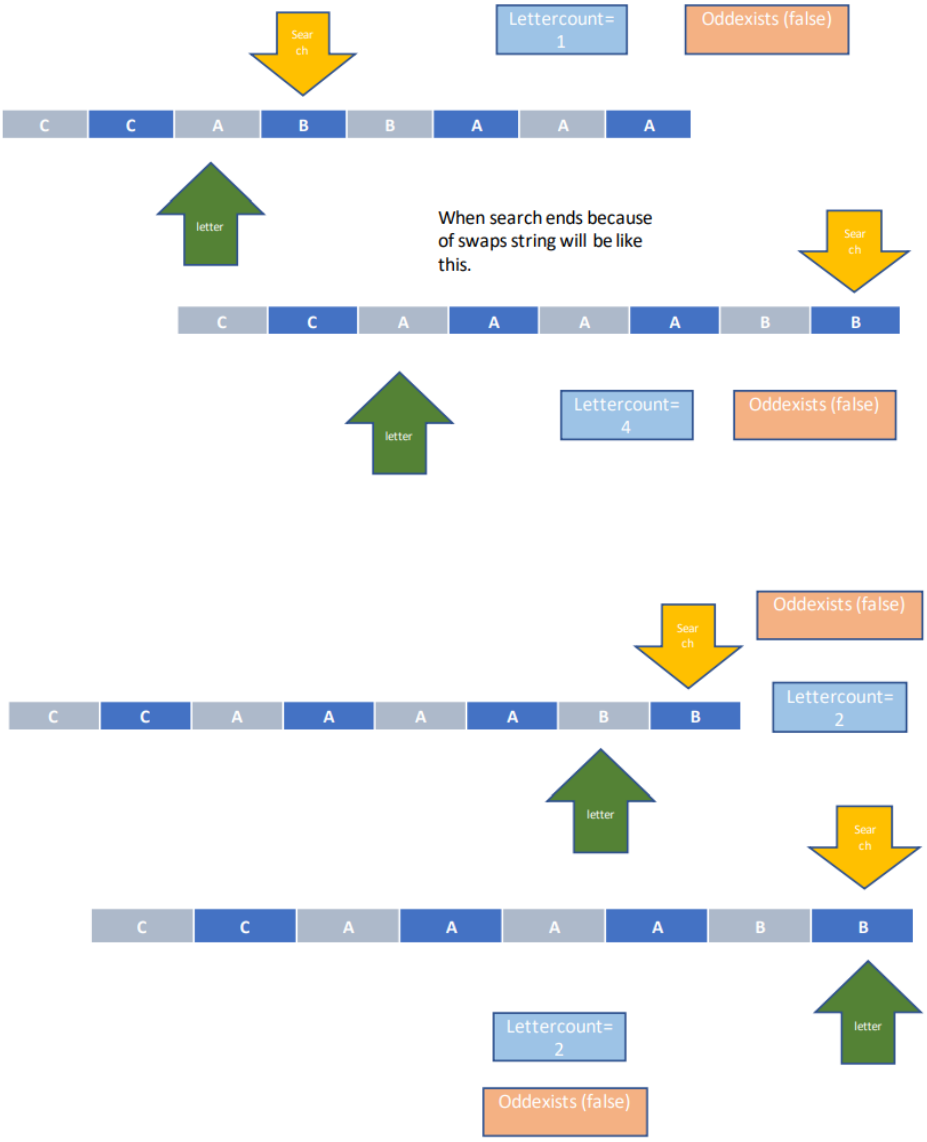
When I looked at palindrome words I realized if a word has every character even amount it can be a palindrome word, or a word has only one odd amount letter and other letters are even amount.

So, my function checks this situation it counts letter and if it is odd amount, it turns boolean oddexists to true. And if oddexists is already true when odd

amount of letter is found. This means that word can’t be a palindrome therefore it returns false.

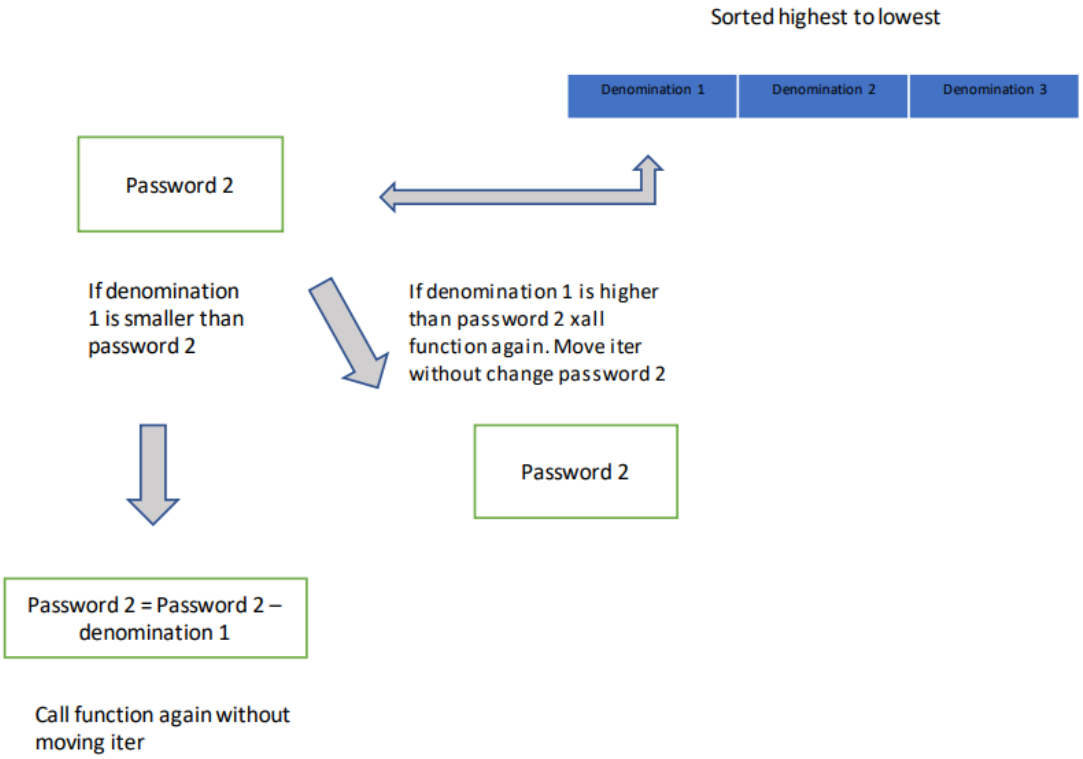
During search of a letter if the same letter is found it is moved next to the letter we search in string.






**Problem solution of isExcactdivision:**

For this function array (will be send to function as a parameter) should be sorted highest to lowest. If the denomination iter is on is smaller than the password function subtracts it from the password and calls itself without moving iterator. If the denomination iter is on is bigger than password function calls itself moving iterator.




## 5.Outputs and run command

## In src/hw4

## Java hw4.security

```
Username : pelin1
Password1 : aa{bb}iii
Password2 : 75
Not a valid username. Username can have only letters
You can't get in
```

```
Username :
Password1 : aa{bb}iii
Password2 : 75
Not a valid username. Username needs at least length 1
```

```
Username :
Password1 : aa{bb}iii
Password2 : 75
Not a valid username. Username needs at least length 1
You can't get in
```

```
Username : pelin
Password1 : aa{bb}iii1
Password2 : 75
Not a valid password1. Password1 can have only letters and parentheses
You can't get in
```

```
Username : pelin
Password1 : aabbiiiiii
Password2 : 75
Not a valid password1. Password1 needs at least 2 bracket
You can't get in
```

```
Username : pelin
Password1 : aa{bbiii
Password2 : 75
Password not balanced
You can't get in
```

```
Username : pelin
Password1 : aabbiii()i
Password2 : 22
password2 is unvalid because denominations can't be summed to a way to make 22
You can't get in
```

```
Username : pelin
Password1 : aa{bb}iiik
Password2 : 75
It is not possible to make a palindrome out of this password
You can't get in
```

```
Username : pelin
Password1 : aa{bb}kkkk
Password2 : 75
This username doesn't contain username spirit
You can't get in
```

```
Username : pelin
Password1 : aabbiiiiii
Password2 : 10000000
Password 2 must be between 10 and  10000
You can't get in

Username : pelin
```

```
Username : pelin
Password1 : aa{bb}iiii
Password2 : 75
The username and passwords are valid. The door is opening, please wait..
```