Submission Worksheet

Submission Data

Course: IT114-450-M2025
Assignment: IT114 Java Problems
Student: Colin R. (ctr26)
Status: Submitted | **Worksheet Progress:** 100+%
Potential Grade: 11.00/10.00 (110.00%)
Received Grade: 0.00/10.00 (0.00%)
Started: 6/6/2025 2:38:01 PM
Updated: 6/7/2025 10:21:52 PM
Grading Link: https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-java-problems/grading/ctr26
View Link: https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-java-problems/view/ctr26

Instructions

- Overview Link: https://youtu.be/Mrahk6SFYao

1. Ensure you read all instructions and objectives before starting.
2. Create a new branch from `main` called `M2-Homework`
   1. `git checkout main` (ensure proper starting branch)
   2. `git pull origin main` (ensure history is up to date)
   3. `git checkout -b M2-Homework` (create and switch to branch)
3. Copy the template code from here: GitHub Repository - M2 Homework
   - It includes Problems 1-4 and a BaseClass. Put all into an `M2` folder or similar (adjust `package` reference at the top if you chose a different folder name).
   - Immediately record to history
     - `git add .`
     - `git commit -m "adding M2 HW baseline files"`
     - `git push origin M2-Homework`
     - Create a Pull Request from `M2-Homework` to `main` and keep it open
4. Fill out the below worksheet
   - Each Problem requires the following as you work
     - Ensure there's a comment with your UCID, date, and brief summary of how the problem was solved
     - Initial outline/plan of how you'll solve it via comments (add/commit after this stage)
     - Code solution (add/commit periodically as needed)
5. Once finished, click "Submit and Export"
6. Locally add the generated PDF to a folder of your choosing inside your repository folder and move it to Github
   1. `git add .`
   2. `git commit -m "adding PDF"`
   3. `git push origin M2-Homework`
   4. On Github merge the pull request from `M2-Homework` to `main`
7. Upload the same PDF to Canvas
8. Sync Local
   1. `git checkout main`

2. `git pull origin main`

Section #1: (2 pts.) Problem 1 - Odds

Progress: 100%

Task #1 (2 pts.) - Edit the `printOdds` method to output odd values of the array

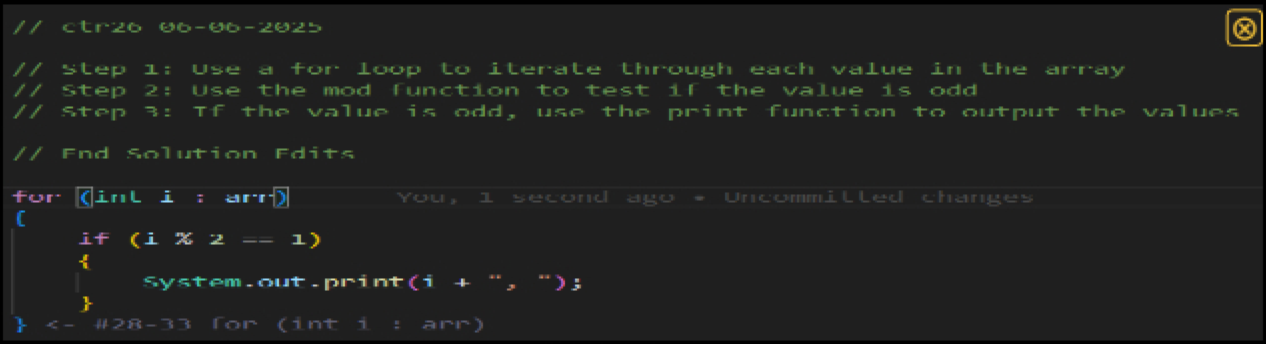
Progress: 100%

Part 1:

Progress: 100%

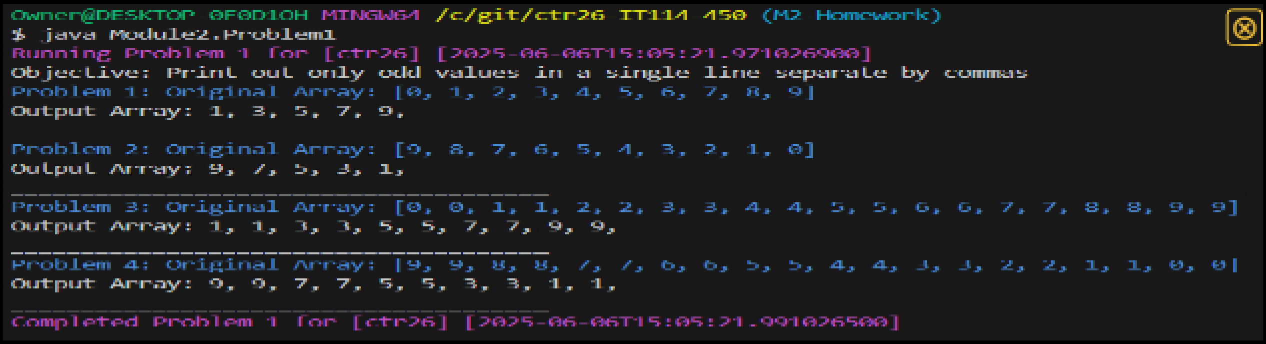
Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program



```
// ctr26 06-06-2025

// Step 1: Use a for loop to iterate through each value in the array
// Step 2: Use the mod function to test if the value is odd
// Step 3: If the value is odd, use the print function to output the values

// End Solution Edits

for (int i : arr)        You, 1 second ago • Uncommitted changes
{
    if (i % 2 == 1)
    {
        System.out.print(i + ", ");
    }
} <- #28-33 for (int i : arr)
```

Relevant code



```
Owner@DESKTOP-0F0D1OH MINGW64 /c/git/ctr26-IT114-450 (M2-Homework)
$ java Module2.Problem1
Running Problem 1 for [ctr26] [2025-06-06T15:05:21.971026900]
Objective: Print out only odd values in a single line separate by commas
Problem 1: Original Array: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
Output Array: 1, 3, 5, 7, 9,

Problem 2: Original Array: [9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
Output Array: 9, 7, 5, 3, 1,
________________________________________
Problem 3: Original Array: [0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9]
Output Array: 1, 1, 3, 3, 5, 5, 7, 7, 9, 9,
________________________________________
Problem 4: Original Array: [9, 9, 8, 8, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]
Output Array: 9, 9, 7, 7, 5, 5, 3, 3, 1, 1,
________________________________________
Completed Problem 1 for [ctr26] [2025-06-06T15:05:21.991026500]
```

Output

Saved: 6/6/2025 3:16:43 PM

Part 2:

Progress: 100%

Details:
Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1
https://github.com/ColinRafferty7/ctr26-IT114-450/M2-Homework/Module2/Problem1.java

URL
https://github.com/ColinRafferty7

Homework/Module2/Problem1.java

Saved: 6/6/2025 3:16:43 PM

Part 3:

Progress: 100%

Details:
Briefly explain how the code solves the challenge (note: this isn't the same as what the code does)

Your Response:

The code solves the problem by checking each value of the array individually, rather than looking at it in it's entirety. This allows the program to check each part for the desired information, which is whether or not the value is odd.

Saved: 6/6/2025 3:16:43 PM

Section #2: (2 pts.) Problem 2 - Sum

Progress: 100%

Task #1 (2 pts.) - Edit the `sumValues` method to sum the array values and present them in a format with exactly two decimal places

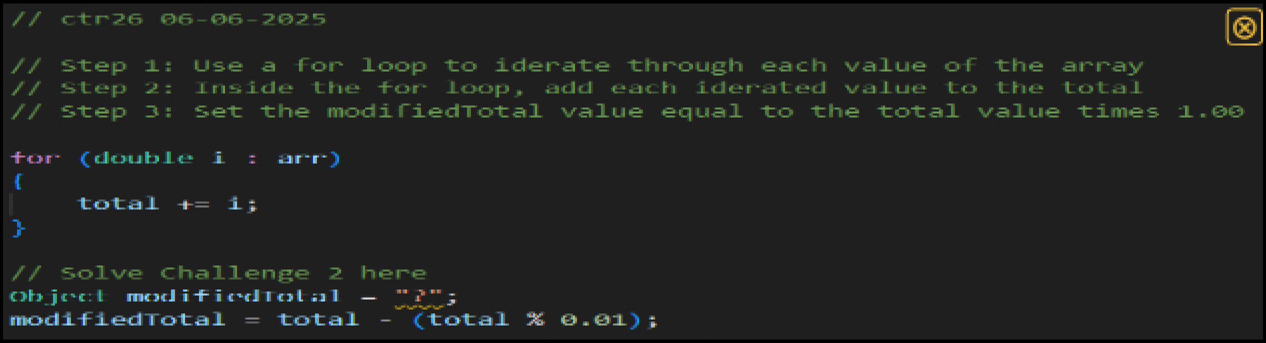
Progress: 100%

Part 1:

Progress: 100%

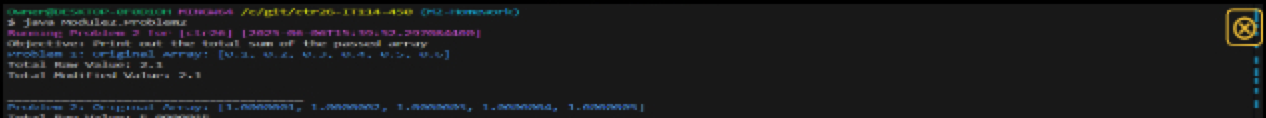
Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

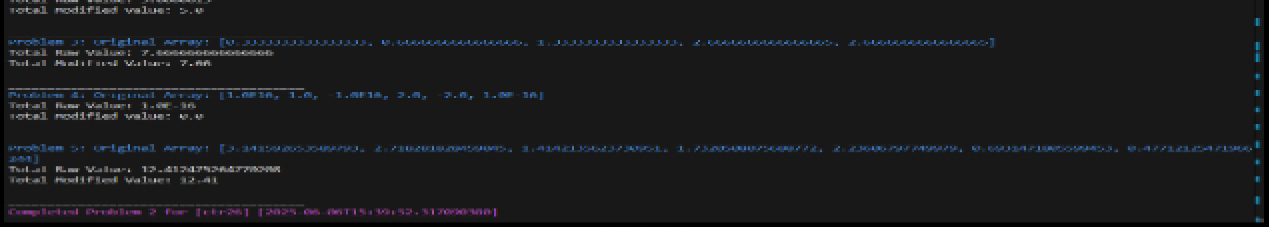


```
// ctr26 06-06-2025

// Step 1: Use a for loop to iderate through each value of the array
// Step 2: Inside the for loop, add each iderated value to the total
// Step 3: Set the modifiedTotal value equal to the total value times 1.00

for (double i : arr)
{
    total += i;
}

// Solve Challenge 2 here
Object modifiedTotal = "?";
modifiedTotal = total - (total % 0.01);
```

Relevant code



```
Owner@[illegible] MINGW64 /c/git/ctr26-IT114-450 (M2-Homework)
$ java Module2.Problem2
Running Problem 2 for [ctr26] [2025-06-06T15:35:52.[illegible]]
Objective: Print out the total sum of the passed array
Problem 1: Original Array: [0.1, 0.2, 0.3, 0.4, 0.5, 0.6]
Total Raw Value: 2.1
Total Modified Value: 2.1

______________________________
Problem 2: Original Array: [1.0000001, 1.0000002, 1.0000003, 1.0000004, 1.0000005]
Total Raw Value: 5.0000015
```



```
Total Modified Value: 5.0

Problem 3: Original Array: [0.3333333333333333, 0.6666666666666666, 1.3333333333333333, 2.6666666666666665, 2.6666666666666665]
Total Raw Value: 7.666666666666666
Total Modified Value: 7.66

______________________________________
Problem 4: Original Array: [1.0E16, 1.0, -1.0E16, 2.0, -2.0, 1.0E-16]
Total Raw Value: 1.0E-16
Total Modified Value: 0.0

Problem 5: Original Array: [3.141592653589793, 2.718281828459045, 1.4142135623730951, 1.7320508075688772, 2.23606797749979, 0.6931471805599453, 0.47712125471966
244]
Total Raw Value: 12.412475264778208
Total Modified Value: 12.41

______________________________________
Completed Problem 2 for [ctr26] [2025-06-06T15:39:52.312090300]
```

Output

Saved: 6/6/2025 3:42:44 PM

Part 2:

Progress: 100%

Details:
Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1
https://github.com/ColinRafferty7/ctr26-IT114-450/M2-Homework/Module2/Problem2.java

URL
https://github.com/ColinRafferty7

Saved: 6/6/2025 3:42:44 PM

Part 3:

Progress: 100%

Details:
Briefly explain how the code solves the challenges (note: this isn't the same as `what` the code does)

Your Response:

The code works by splitting the array into its individual parts to make it easy to handle. Then, it adds all of the values to a total variable that stores them. Then, it does math operations on the total value and sets it equal to a new modifiedTotal variable.

Saved: 6/6/2025 3:42:44 PM

Section #3: (2 pts.) Problem 3 - Conversion

Progress: 100%

Task #1 (2 pts.) - Edit the `bePositive` method to make each value positive, convert it back to the orginal data type, and set it to the proper slot in the `output` array

Progress: 100%

Part 1:

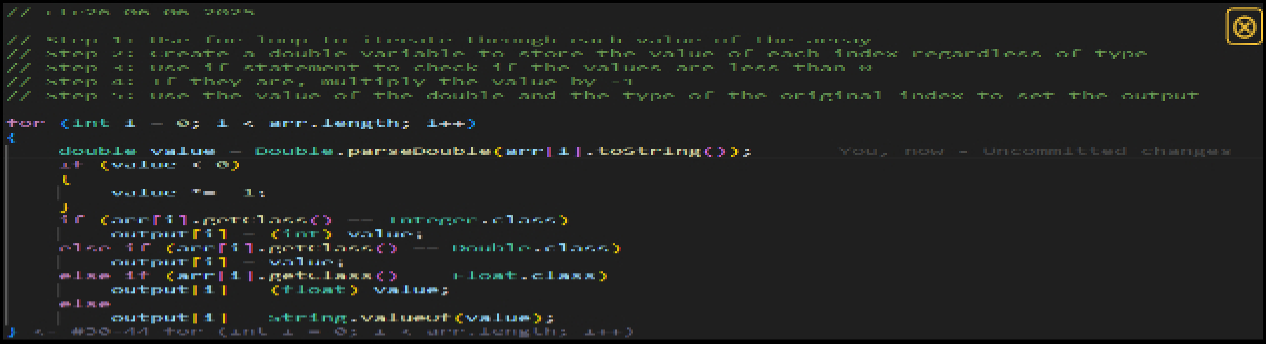
Progress: 100%

Progress: 100%

Details:

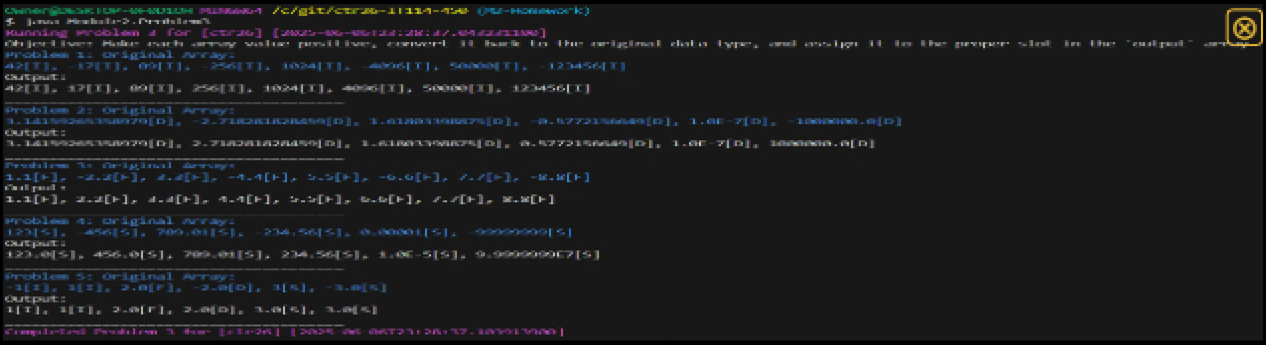
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program



```
// ctr26 06 06 2025

// Step 1: Use for loop to iterate through each value of the array
// Step 2: create a double variable to store the value of each index regardless of type
// Step 3: use if statement to check if the values are less than 0
// Step 4: if they are, multiply the value by -1
// Step 5: use the value of the double and the type of the original index to set the output

for (int i = 0; i < arr.length; i++)
{
    double value = Double.parseDouble(arr[i].toString());        You, now - Uncommitted changes
    if (value < 0)
    {
        value *= -1;
    }
    if (arr[i].getClass() == Integer.class)
        output[i] = (int) value;
    else if (arr[i].getClass() == Double.class)
        output[i] = value;
    else if (arr[i].getClass() == Float.class)
        output[i] = (float) value;
    else
        output[i] = String.valueOf(value);
} <- #30-44 for (int i = 0; i < arr.length; i++)
```

Code



```
Owner@DESKTOP-0H0D1DH MINGW64 /c/git/ctr26-IT114-450 (M2-Homework)
$ java Module2.Problem3
Running Problem 3 for [ctr26] [2025-06-06T23:28:37.044223100]
Objective: Make each array value positive, convert it back to the original data type, and assign it to the proper slot in the `output` array
Problem 1: Original Array:
42[I], -17[I], 99[I], -256[I], 1024[I], -4096[I], 50000[I], -123456[I]
Output:
42[I], 17[I], 99[I], 256[I], 1024[I], 4096[I], 50000[I], 123456[I]
------------------------------------------------------------
Problem 2: Original Array:
3.14159265358979[D], -2.71828182845905[D], 1.61803398875[D], -0.5772156649[D], 1.0E-7[D], -1000000.0[D]
Output:
3.14159265358979[D], 2.71828182845905[D], 1.61803398875[D], 0.5772156649[D], 1.0E-7[D], 1000000.0[D]
------------------------------------------------------------
Problem 3: Original Array:
1.1[F], -2.2[F], 3.3[F], -4.4[F], 5.5[F], -6.6[F], 7.7[F], -8.8[F]
Output:
1.1[F], 2.2[F], 3.3[F], 4.4[F], 5.5[F], 6.6[F], 7.7[F], 8.8[F]
------------------------------------------------------------
Problem 4: Original Array:
123[S], -456[S], 789.01[S], -234.56[S], 0.00001[S], -99999999[S]
Output:
123.0[S], 456.0[S], 789.01[S], 234.56[S], 1.0E-5[S], 9.9999999E7[S]
------------------------------------------------------------
Problem 5: Original Array:
-1[I], 1[I], 2.0[F], -2.0[D], 3[S], -3.0[S]
Output:
1[I], 1[I], 2.0[F], 2.0[D], 3.0[S], 3.0[S]
------------------------------------------------------------
Completed Problem 3 for [ctr26] [2025-06-06T23:28:37.103913000]
```

Output

Saved: 6/6/2025 11:36:55 PM

Part 2:

Progress: 100%

Details:

Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1

https://github.com/ColinRafferty7/ctr26-IT114-450/M2-Homework/Module2/Problem3.java

URL

https://github.com/ColinRafferty7

Saved: 6/6/2025 11:36:55 PM

Part 3:

Progress: 100%

Details:

Briefly explain `how` the code solves the challenges (note: this isn't the same as `what` the code does)

Your Response:

Your Response:

The code works by first seperating the value from any unique object. All of the values are copied to a new double variable to be able to run math functions. By seperating the value from the type, the program can now go back to the original value and use its type in order to convert the newly changed double value.

Saved: 6/6/2025 11:36:55 PM

Section #4: (2 pts.) Problem 4 - Strings

Progress: 100%

Task #1 (2 pts.) - Edit the `transformText` method to solve the challenges

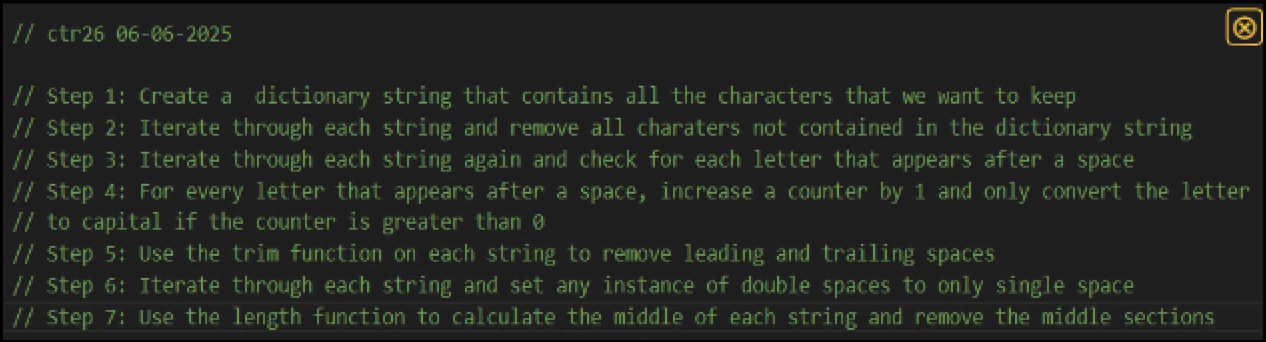
Progress: 100%

Part 1:

Progress: 100%

Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

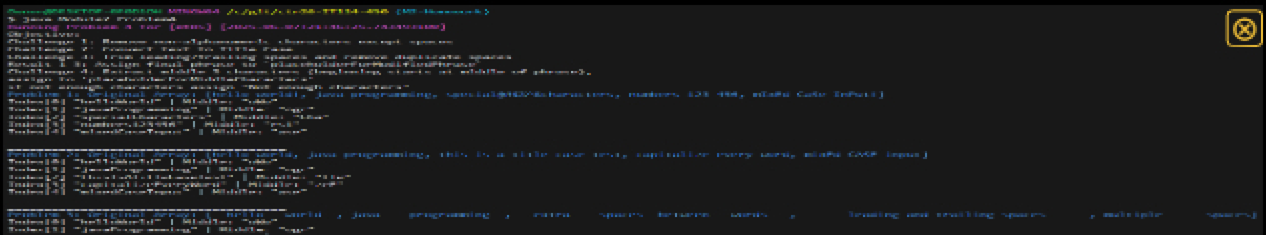


```
// ctr26 06-06-2025

// Step 1: Create a  dictionary string that contains all the characters that we want to keep
// Step 2: Iterate through each string and remove all charaters not contained in the dictionary string
// Step 3: Iterate through each string again and check for each letter that appears after a space
// Step 4: For every letter that appears after a space, increase a counter by 1 and only convert the letter
// to capital if the counter is greater than 0
// Step 5: Use the trim function on each string to remove leading and trailing spaces
// Step 6: Iterate through each string and set any instance of double spaces to only single space
// Step 7: Use the length function to calculate the middle of each string and remove the middle sections
```

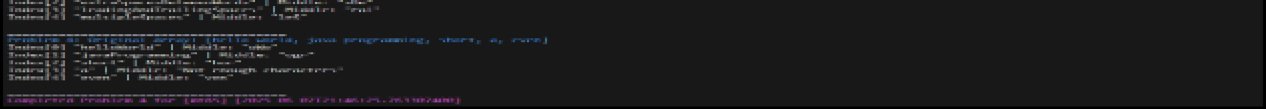
Outline

```
[illegible]
```

Code



```
[illegible]
```



```
Index[2] [illegible] | Middle: [illegible]
Index[3] "leadingAndTrailingSpaces" | Middle: [illegible]
Index[4] "multipleSpaces" | Middle: [illegible]

==========================================
Problem 4: Original Array: [hello world, java programming, short, a, even]
Index[0] "helloWorld" | Middle: [illegible]
Index[1] "javaProgramming" | Middle: [illegible]
Index[2] "short" | Middle: [illegible]
Index[3] "a" | Middle: "Not enough characters"
Index[4] "even" | Middle: [illegible]

==========================================
Completed Problem 4 for [illegible] [illegible]
```

Output

Saved: 6/7/2025 10:03:25 PM

Part 2:

Progress: 100%

Details:
Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1
https://github.com/ColinRafferty7/ctr26-IT114-450-M2-Homework/Module2/Problem4.java

URL
https://github.com/ColinRafferty7

Saved: 6/7/2025 10:03:25 PM

Part 3:

Progress: 100%

Details:
Briefly explain how the code solves the challenges (note: this isn't the same as what the code does)

Your Response:

The code solves the problem by tackling each challenge seperately. By splitting the overall goal into many small pieces, it makes it much easier to handle each change. It also functions by continuously altering the contents of the arr[i] variable with each new change to the string.

Saved: 6/7/2025 10:03:25 PM

Task #2 (+ 1.11 pts.) - Edit the `transformText` method to solve the extra credit challenge (challenge 4)

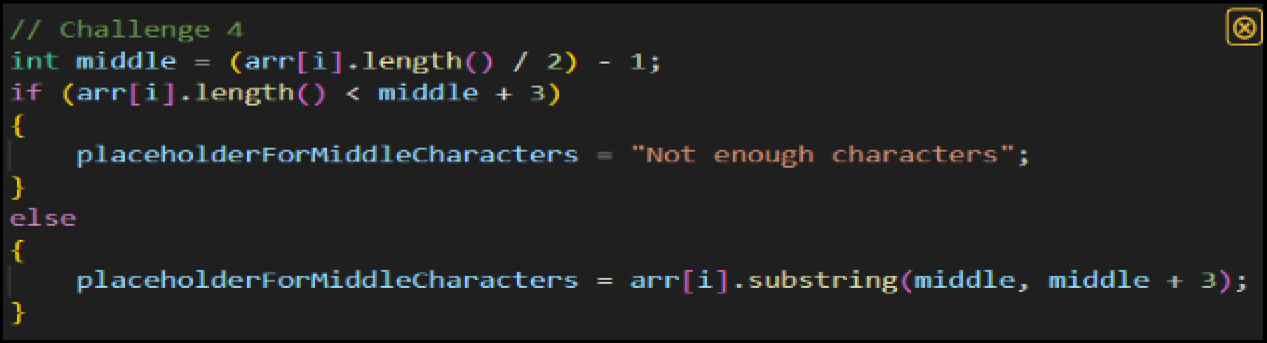
Progress: 100%

Part 1:

Progress: 100%

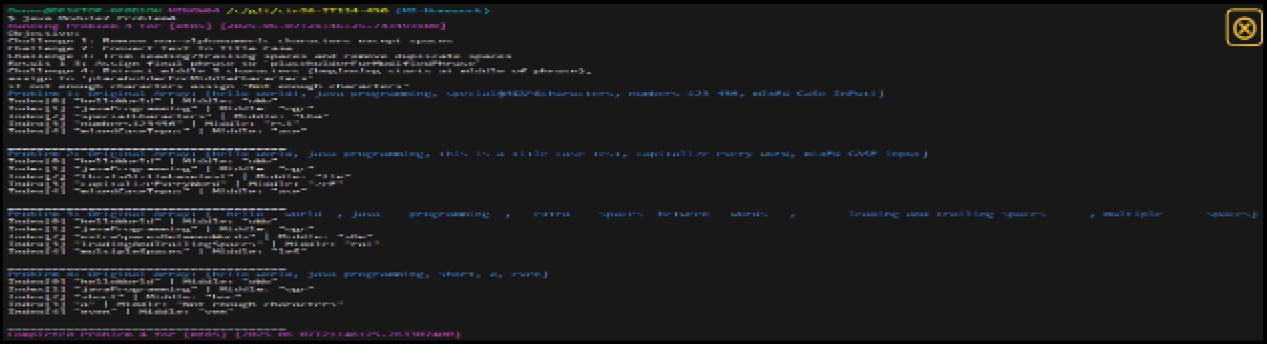
Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program



```
// Challenge 4
int middle = (arr[i].length() / 2) - 1;
if (arr[i].length() < middle + 3)
{
    placeholderForMiddleCharacters = "Not enough characters";
}
else
{
    placeholderForMiddleCharacters = arr[i].substring(middle, middle + 3);
}
```

Code



```
[illegible]
$ java [illegible]
Running Problem 4 for [illegible] [illegible]
Objectives
Challenge 1: Remove non-alphanumeric characters except spaces
Challenge 2: Convert text to Title Case
Challenge 3: Trim leading/trailing spaces and remove duplicate spaces
Result 1-3: Assign final phrase to "placeholderForModifiedPhrase"
Challenge 4: Extract middle 3 characters (beginning starts at middle of phrase),
assign to "placeholderForMiddleCharacters"
if not enough characters assign "Not enough characters"
Problem 1: Original Array: [hello world!, java programming, special@#$%^&characters, number 123 456, mIxEd CaSe InPut]
Index[0] "helloWorld" | Middle: "oWo"
Index[1] "javaProgramming" | Middle: "ogr"
Index[2] "specialCharacters" | Middle: "Cha"
Index[3] "number123456" | Middle: "r12"
Index[4] "mixedCaseInput" | Middle: "Cas"

============================================================
Problem 2: Original Array: [hello world, java programming, this is a title case test, capitalize every word, mIxEd CASE input]
Index[0] "helloWorld" | Middle: "oWo"
Index[1] "javaProgramming" | Middle: "ogr"
Index[2] "thisIsATitleCaseTest" | Middle: "tle"
Index[3] "capitalizeEveryWord" | Middle: "zeE"
Index[4] "mixedCaseInput" | Middle: "Cas"

============================================================
Problem 3: Original Array: [  hello   world  , java    programming  ,  extra    spaces  between   words  ,      leading and trailing spaces      , multiple      spaces]
Index[0] "helloWorld" | Middle: "oWo"
Index[1] "javaProgramming" | Middle: "ogr"
Index[2] "extraSpacesBetweenWords" | Middle: "sBe"
Index[3] "leadingAndTrailingSpaces" | Middle: "Tra"
Index[4] "multipleSpaces" | Middle: "leS"

============================================================
Problem 4: Original Array: [hello world, java programming, short, a, even]
Index[0] "helloWorld" | Middle: "oWo"
Index[1] "javaProgramming" | Middle: "ogr"
Index[2] "short" | Middle: "hor"
Index[3] "a" | Middle: "Not enough characters"
Index[4] "even" | Middle: "ven"

============================================================
Completed Problem 4 for [illegible] [illegible]
```

Output

Saved: 6/7/2025 10:12:09 PM

Part 2:

Progress: 100%

Details:
Briefly explain how the code solves the extra credit challenge (note: this isn't the same as what the code does)

Your Response:

The code solves the extra credit challenge by first determining the center of the string with some math functions, and then verifying if the string is long enough to create a valid output. If it is, then a substring will be created and copied to the output.

Saved: 6/7/2025 10:12:09 PM

Section #5: (2 pts.) Misc

Progress: 100%

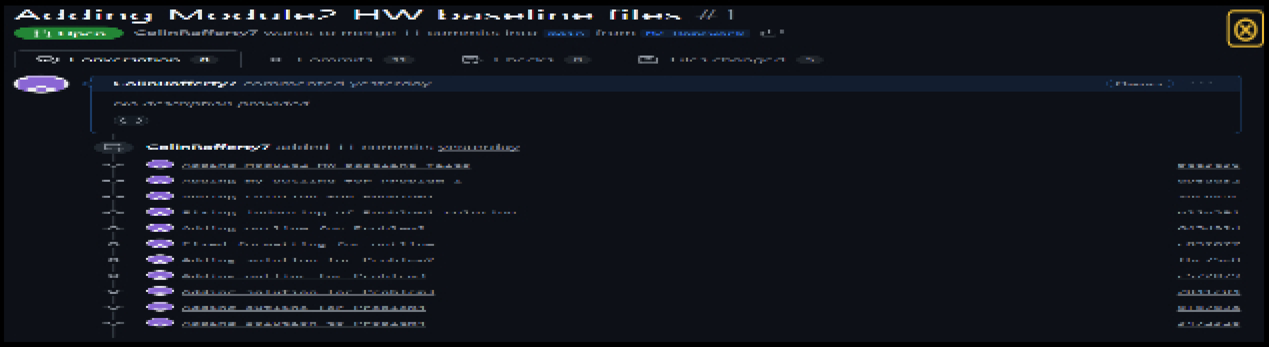
Task #1 (0.67 pts.) - Github Details

Progress: 100%

Part 1:

Progress: 100%

Details:
From the Commits tab of the Pull Request screenshot the commit history Following minimum should be present



Adding Module2 HW baseline files #1

Open ColinRafferty7 wants to merge 11 commits into main from M2-Homework

ColinRafferty7 commented yesterday

No description provided

ColinRafferty7 added 11 commits yesterday

[illegible]

Commit history

Saved: 6/7/2025 10:13:49 PM

Part 2:

Progress: 100%

Details:
Include the link to the Pull Request (should end in `/pull/#`)

URL #1
https://github.com/ColinRafferty7/ctr26-IT114-450/pull/1

URL
https://github.com/ColinRafferty7

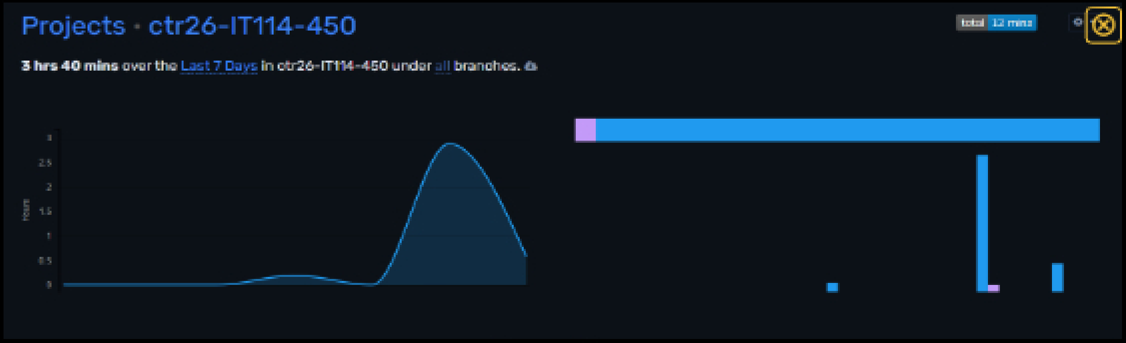
Saved: 6/7/2025 10:13:49 PM

Task #2 (0.67 pts.) - WakaTime - Activity

Progress: 100%

Details:

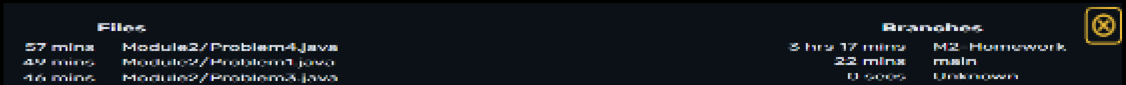
- Visit the WakaTime.com Dashboard
- Click `Projects` and find your repository
- Capture the overall time at the top that includes the repository name
- Capture the individual time at the bottom that includes the file time
- Note: The duration isn't relevant for the grade and the visual graphs aren't necessary



Projects · ctr26-IT114-450

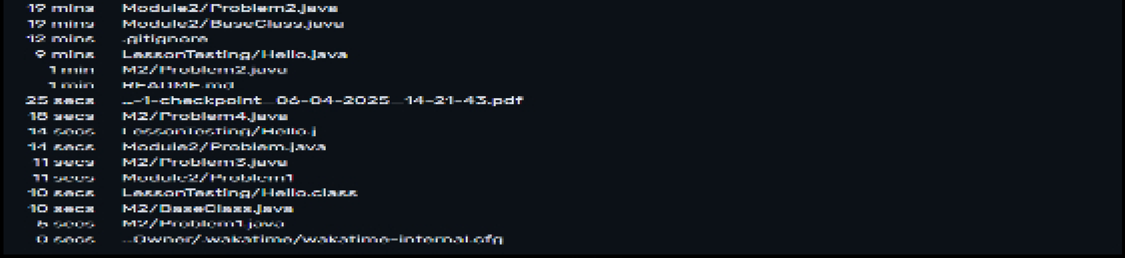
3 hrs 40 mins over the Last 7 Days in ctr26-IT114-450 under all branches.

Total time



Files	
57 mins	Module2/Problem4.java
49 mins	Module2/Problem1.java
46 mins	Module2/Problem3.java

Branches	
3 hrs 17 mins	M2-Homework
22 mins	main
0 secs	Unknown



```
 12 mins   Module2/Problem2.java
 12 mins   Module2/BaseClass.java
 12 mins   .gitignore
  9 mins   LessonTesting/Hello.java
   1 min   M2/Problem2.java
   1 min   README.md
 25 secs   ...-1-checkpoint__06-04-2025__14-21-43.pdf
 18 secs   M2/Problem4.java
 14 secs   LessonTesting/Hello.j
 14 secs   Module2/Problem.java
 11 secs   M2/Problem3.java
 11 secs   Module2/Problem1
 10 secs   LessonTesting/Hello.class
 10 secs   M2/BaseClass.java
  5 secs   M2/Problem1.java
  0 secs   ...Owner/.wakatime/wakatime-internal.cfg
```

File times

Saved: 6/7/2025 10:15:04 PM

Task #3 (0.67 pts.) - Reflection

Progress: 100%

Task #1 (0.33 pts.) - What did you learn?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

In the assignment, I relearned much of the Java language that I had forgotten since taking CS113. Going into the problem, I had an idea for what was possible to do with the language, but needed to work out the ways in which I was able to do it. This was good practice to remove some of the rust that I have for Java by challenging me to make many changes to many different data types.

Saved: 6/7/2025 10:17:02 PM

Task #2 (0.33 pts.) - What was the easiest part of the assignment?

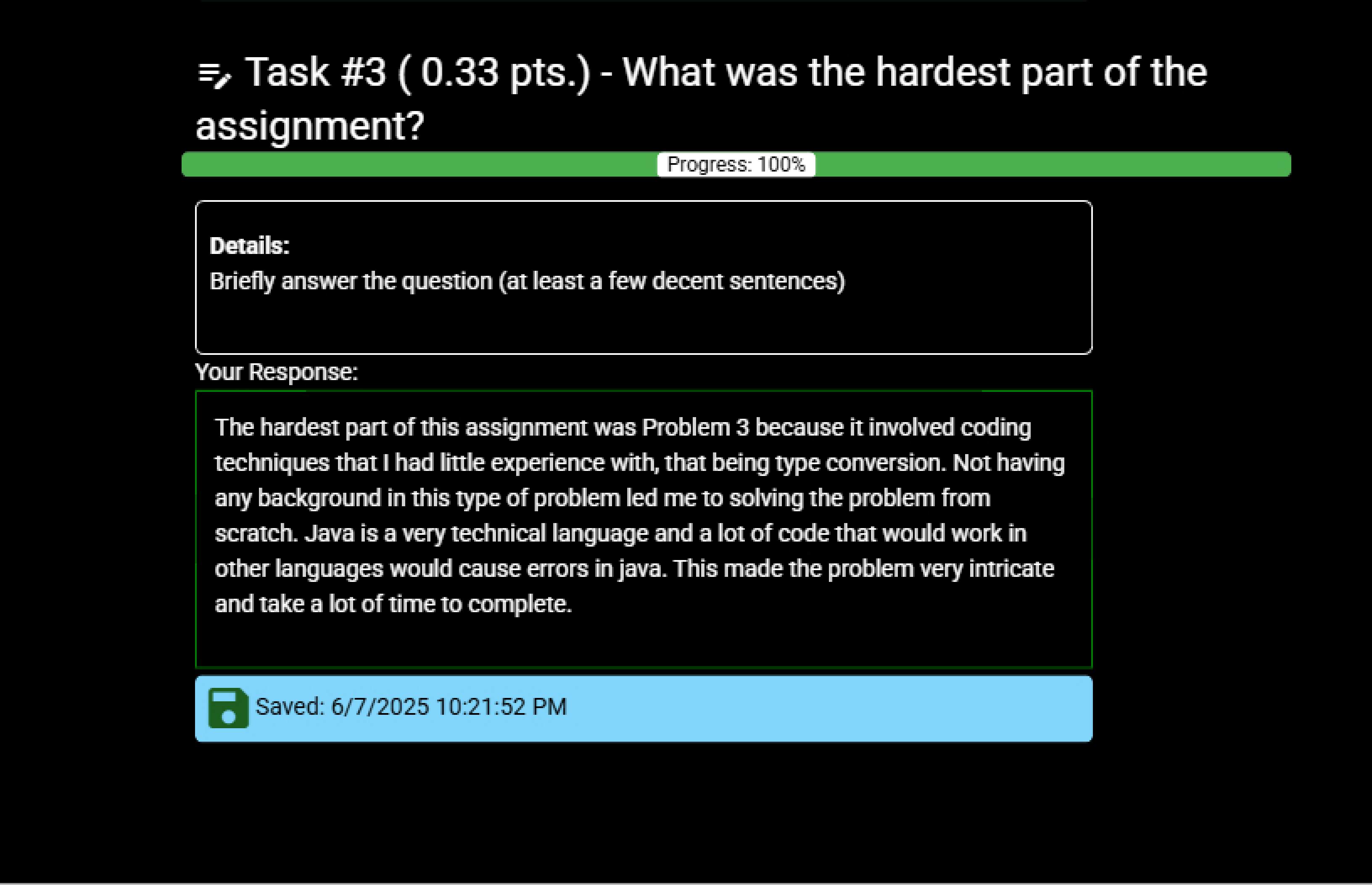
Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The easiest part of this assignment was Problem 1 which I was instantly able to solve. It contained very basic coding techniques that transfer over from all of the other languages I work with. Regardless, it was a good warmup for the rest of the assignment.

Saved: 6/7/2025 10:19:21 PM



Task #3 (0.33 pts.) - What was the hardest part of the assignment?

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The hardest part of this assignment was Problem 3 because it involved coding techniques that I had little experience with, that being type conversion. Not having any background in this type of problem led me to solving the problem from scratch. Java is a very technical language and a lot of code that would work in other languages would cause errors in java. This made the problem very intricate and take a lot of time to complete.

Saved: 6/7/2025 10:21:52 PM