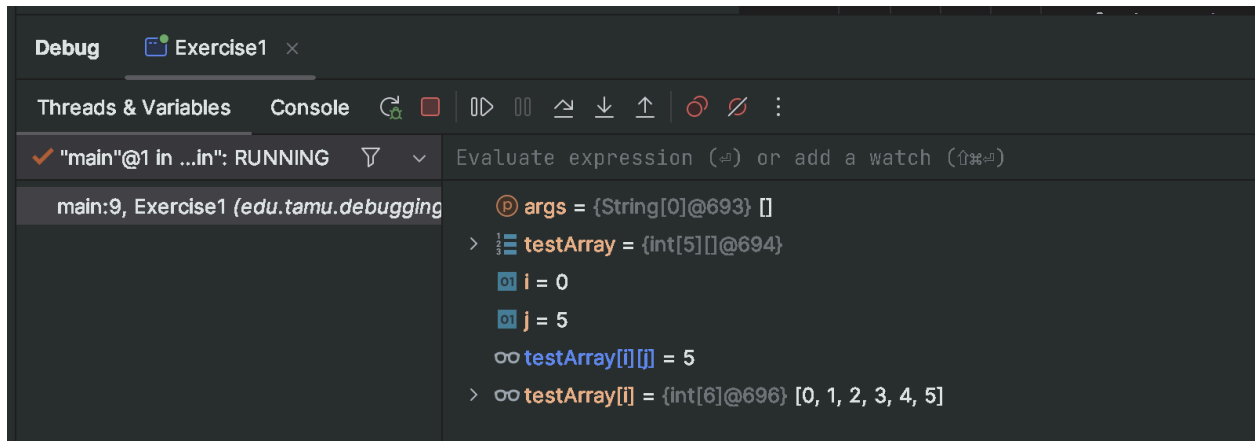
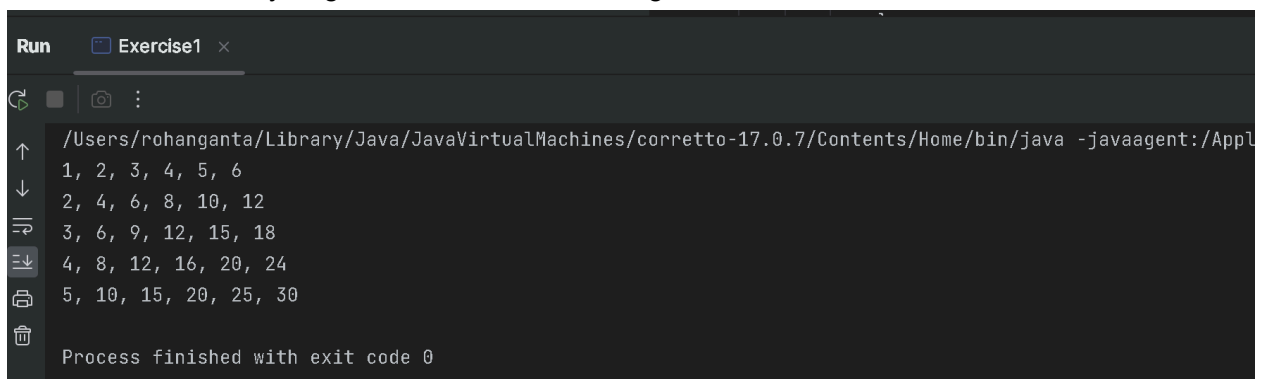


Exercise 1:

1. Done
2. $i=0, j=5$



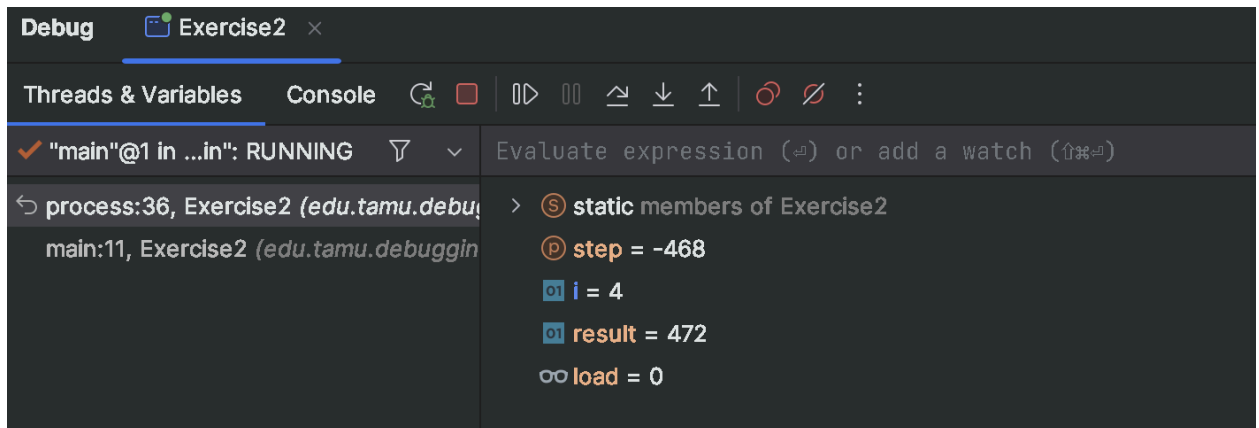
3. No, it is not correct because the inner loop goes from 1 to 6, skipping the value of index 0 in the array. Therefore $(i+1)*j$ will not be valid because it will not initialize the values in the correct location, and when it is in the right location, we will need to do $j + 1$ so that values in the array do not begin with 0.
4. The error is that the inner loop goes from 1-6 rather than 0-5, which skips initializing the first value, and will try to go to the index out of range.



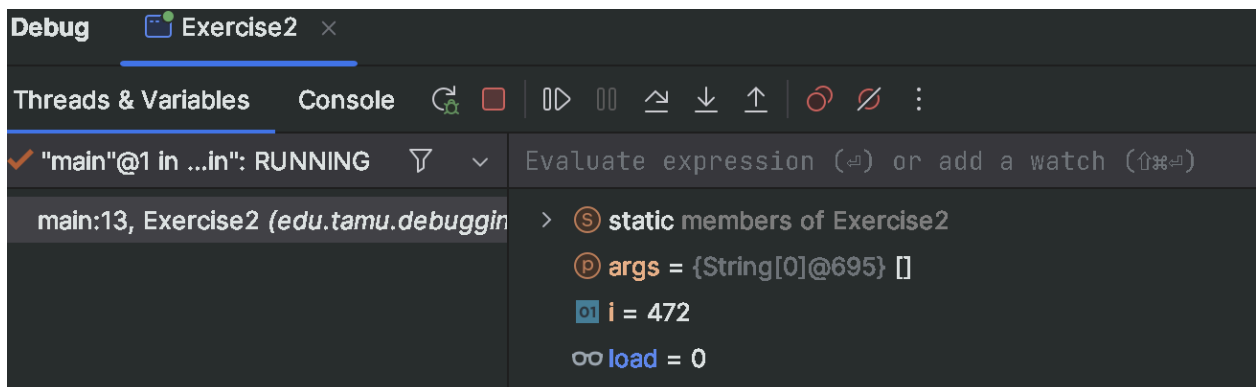
5.

Exercise 2.

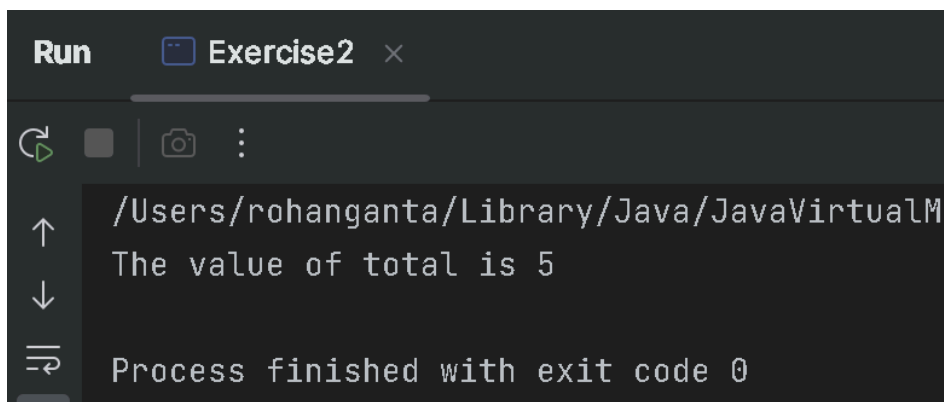
1. $i=4$, $step=-468$, $result=472$, $load=0$



2. Load=0

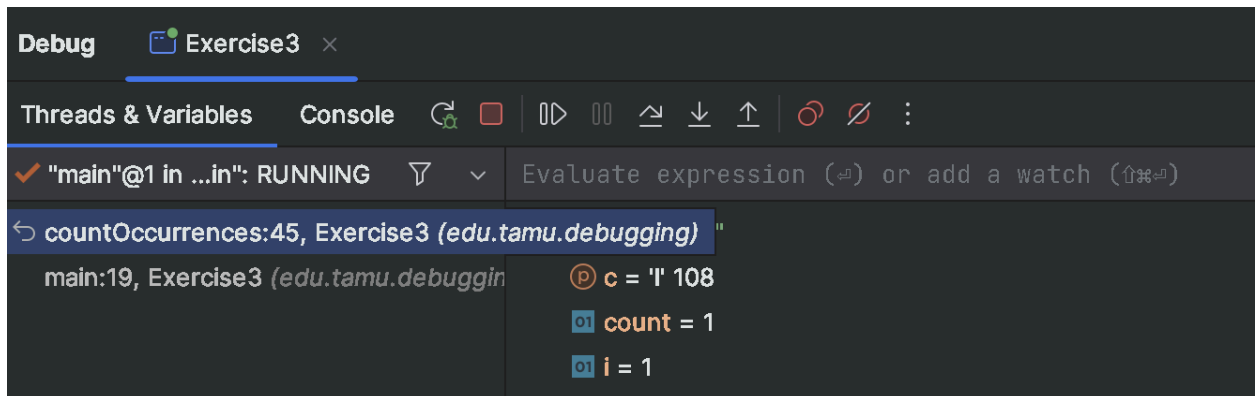


3. It crashes because $load=0$, and the line divides $i*100$ by $load$. This leads to a divide-by-zero error.
4. The error was that we don't check if $load$ is 0 before dividing by it, which leads to an error when it is 0. I fixed it by doing the division in the else section of the if-else block for only when $load$ is not 0.
- 5.



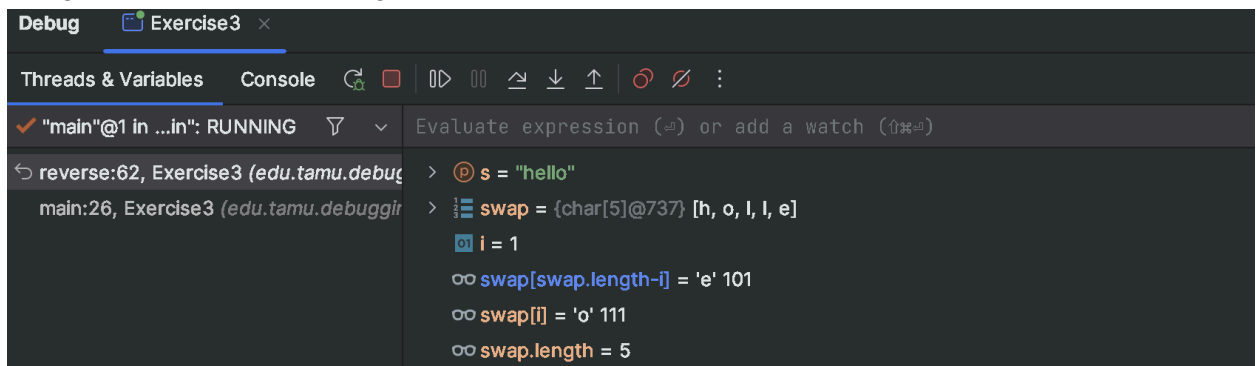
Exercise 3:

1. $i = 1$ and $\text{count} = 1$ at first iteration, meaning that before the string is checked, it already believes it has found one occurrence due to count being 1, and will skip the first letter due to i being 1.



```
Debug Exercise3 x
Threads & Variables Console
"main"@1 in ...in": RUNNING
countOccurrences:45, Exercise3 (edu.tamu.debugging)
main:19, Exercise3 (edu.tamu.debugging)
c = 'l' 108
count = 1
i = 1
```

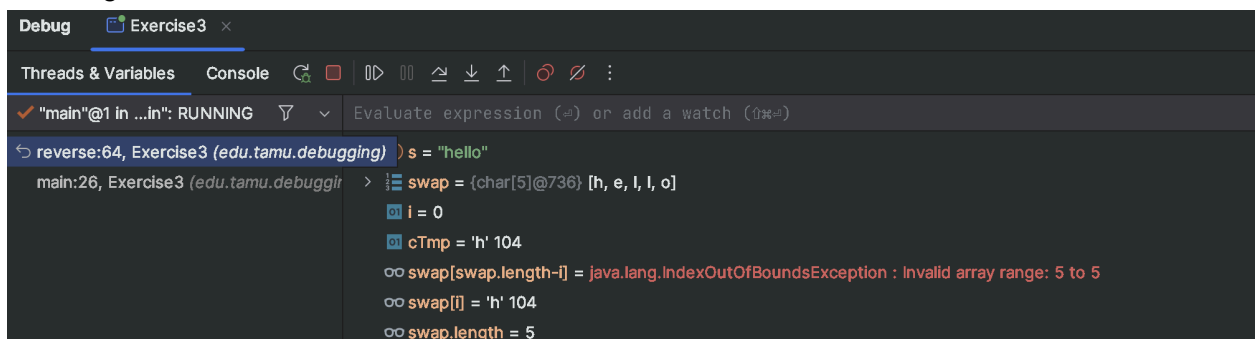
2. It begins at index 1, swapping last element with second instead of first



```
Debug Exercise3 x
Threads & Variables Console
"main"@1 in ...in": RUNNING
reverse:62, Exercise3 (edu.tamu.debugging)
main:26, Exercise3 (edu.tamu.debugging)
s = "hello"
swap = {char[5]@737} [h, o, l, l, e]
i = 1
swap[swap.length-i] = 'e' 101
swap[i] = 'o' 111
swap.length = 5
```

Therefore, I switched i to 0 and switched \leq to just $<$.

Then I got index out of bounds

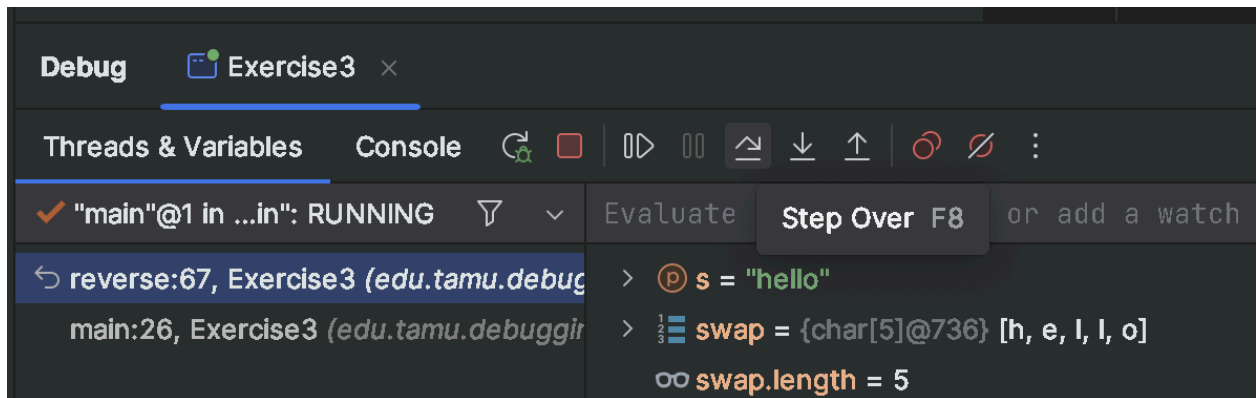


```
Debug Exercise3 x
Threads & Variables Console
"main"@1 in ...in": RUNNING
reverse:64, Exercise3 (edu.tamu.debugging)
main:26, Exercise3 (edu.tamu.debugging)
s = "hello"
swap = {char[5]@736} [h, e, l, l, o]
i = 0
cTmp = 'h' 104
swap[swap.length-i] = java.lang.IndexOutOfBoundsException: Invalid array range: 5 to 5
swap[i] = 'h' 104
swap.length = 5
```

This is because we are subtracting i , which is 0-index based, from swap.length , which is 1 index based. To fix this, we subtract 1 as well.

The second bug is that we iterate through the entire string, but it reverses after iterating through half. Therefore, if we iterate through the second half, it will reverse again and

unreverse the string.



In my debugging console, I saw the string unreversing.

3. The errors in method 1 was that `i` began at 1 instead of 0, and count began at 1 instead of 0. The errors in method 2 is that `i` in the loop began at 1 instead of 0, which had to be fixed and the other error was that the loop went through the whole string which would unreverse it.
4. In method 1, I switched count from 1 to 0 and `i` from 1 to 0. In method 2, I switched the loop from 1 to `swap.length` to 0 to `swap.length/2`. I also subtracted 1 from `swap.length` within the function so that it would be index out of bounds.
- 5.

