

M. Hellard, J. Laframboise, C. Miller  
B. Patrick  
COIS 2020H  
October 16th 2022

## Assignment 1 - Testing

### PART A:

Scenario 1: size 0, with no black squares.

```
Input size of crossword :0
Input number of black spaces :0
*Note: Colours inverted*
-
Across:      Down:
Symmetric
```

Scenario 2: size negative, with negative black squares.

The screenshot shows a C# code editor with a file named 'Puzzle.cs'. The code defines a 'Puzzle' class with a private integer 'N' and a public constructor 'Puzzle(int N)'. The constructor includes a check for negative values: if (N < 0) { throw new Exception("Cannot construct Puzzle with negative value"); }. The code is at line 38, where the exception is thrown. A tooltip for 'Exception Unhandled' is visible, showing the message 'System.Exception: 'Cannot construct Puzzle with negative value''. The status bar at the bottom indicates 'No issues found' and 'Ln: 38 Ch: 13 SPC CRLF'.

```
31 private int N;  
32  
33 // Create an NxN crossword grid of WHITE squares (4 marks)  
34 public Puzzle(int N)  
35 {  
36     if (N < 0)  
37     {  
38         throw new Exception("Cannot construct Puzzle with negative value");  
39     }  
40     this.N = N;  
41     grid = new Square[this.N, this.N];  
42     for (int ix = 0; ix < this.N; ix++)  
43     {  
44         for (int iy = 0; iy < this.N; iy++)  
45             grid[ix, iy] = new Square(); //see Square constructor  
46     }  
47 }
```

Scenario 3: size positive, number of black squares less than size.

The screenshot shows a crossword puzzle grid with input size 4 and 1 black space. The grid is 4x4. The input size is 4, and the input number of black spaces is 1. The note says 'Colours inverted'. The grid is shown with black squares (###) and white squares ( ). The grid is as follows:

1	###	2	3
	###		
	###		
4	5		
6			
7			

Across: 2, 4, 6, 7  
Down: 1, 2, 3, 5

Not Symmetric

Scenario 4: size positive, number of black squares greater than size.

```
Input size of crossword :2
Input number of black spaces :30
*Note: Colours inverted*
```

```
-----
|##|##|
|##|##|
-----
|##|##|
|##|##|
-----
```

Across:        Down:

Symmetric

Symmetry test

```
Input size of crossword :2
Input number of black spaces :4
*Note: Colours inverted*
```

```
-----
|##|##|
|##|##|
-----
|##|##|
|##|##|
-----
```

Across:        Down:

Symmetric

## Size of 1 test

```
Input size of crossword :1
Input number of black spaces :1
*Note: Colours inverted*
```

```
----
|##|
|##|
----
```

Across:        Down:

Symmetric

## PART B:

### *Creating a string*

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

After entering “1”...

```
Please type the string you would like to create.  
ABCD
```

*Reversing the string*

```
Part B Main Menu  
1: Create a string  
2: Reverse a string  
3: Return the index of the first occurrence of a char in a string  
4: Remove a character from a string  
5: Check the equality of an object to MyString  
6: Print a MyString object  
7: Quit  
Please type in the number of the method you want to use.
```

After entering “2”...

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
```

After entering the position of the string, in this case, “1”...

```
D
C
B
A
```

### *Returning the index of the first occurrence of a character in a string*

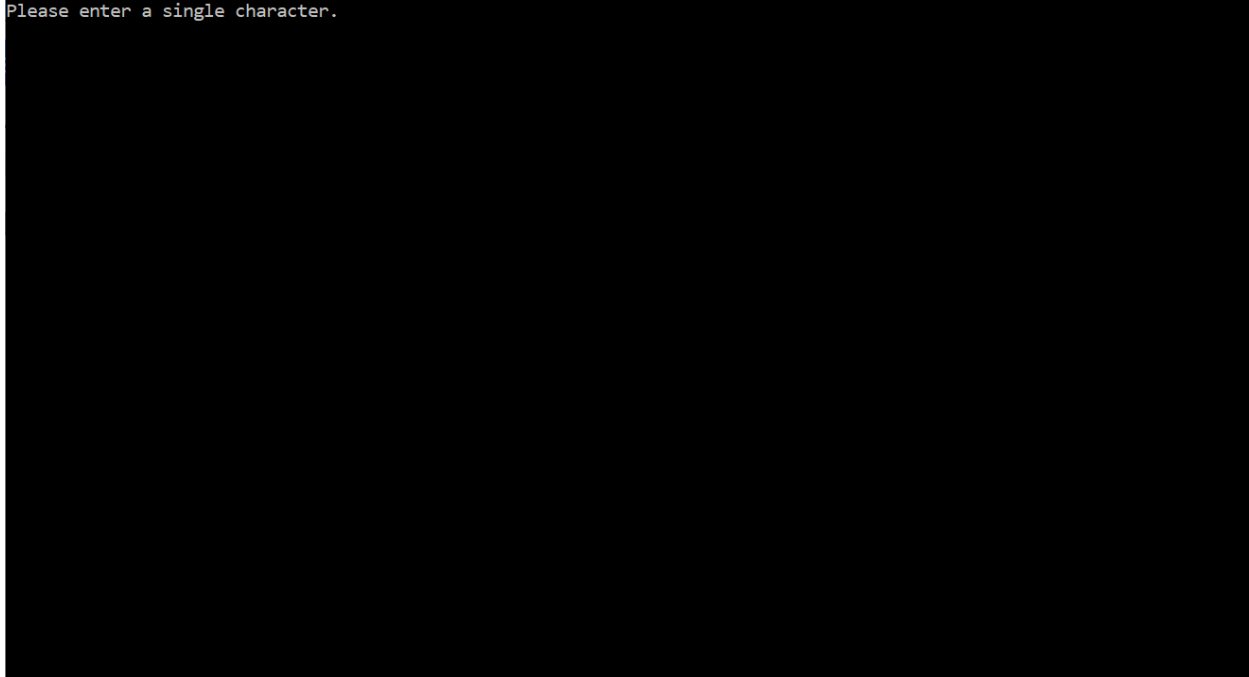
```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

After entering “3”...

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
```

After entering the position of the string, in this case, “1”...

```
Please enter a single character.
```



After entering the desired character, in this case, “A”...

```
Please enter a single character.
```

```
A  
4
```





In the event that an unused character is attempted, a return to the main menu is implemented

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

### *Removing a character from the string*

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

After entering “4”...

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
```

After entering the position of the string, in this case, “1”...

```
Please enter a single character.
```

After entering the desired character, in this case, “A”...

D  
C  
B

In the event that an unused character is attempted, a return to the main menu is implemented

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

### *Checking the equality of an object to MyString*

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

### After entering "5"...

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
```

After entering the position of the string, in this case, “1”...

```
Compare against a user created string or another object?  
1: User String  
2: A non-MyString Object
```

After entering “1”...

```
0: Program+MyString  
1: Program+MyString  
2:  
3:  
4:  
5:  
6:  
7:  
8:  
9:  
10:  
11:  
12:  
13:  
14:  
15:  
16:  
17:  
18:  
19:  
Please type the number to select the Object.  
1
```

After entering the position of the string for comparison, in this case, “0”...

```
Compare against a user created string or another object?  
1: User String  
2: A non-MyString Object
```

After entering the position of the string, in this case, “1”...

```
0: Program+MyString  
1: Program+MyString  
2:  
3:  
4:  
5:  
6:  
7:  
8:  
9:  
10:  
11:  
12:  
13:  
14:  
15:  
16:  
17:  
18:  
19:  
Please type the number to select the Object.  
1  
False
```

Comparing string 1 with itself returns a positive result

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
1
True
```

*Returning the string*

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
```

After entering “6”...

```
0: Program+MyString
1: Program+MyString
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
Please type the number to select the Object.
```

After entering the position of the string, in this case, “1”...

```
A
B
C
D
```



*Entering an invalid selection, in this case, “14”, results in no advancement. Non-numerical keys result in the same progression.*

```
Part B Main Menu
1: Create a string
2: Reverse a string
3: Return the index of the first occurrence of a char in a string
4: Remove a character from a string
5: Check the equality of an object to MyString
6: Print a MyString object
7: Quit
Please type in the number of the method you want to use.
14
```