

1) Write application that creates an array of double, to provide following functionality.

display the length of the array and its elements.

Display an array. (Use for each version of loop for display).

compute the sum of the squares of these numbers.

Determine Mean and Median of an array.

Sort an array – Ascending and Descending. Use any two sorting algorithm. User can also select the sorting method.

Search an element from the array, i.e. returns the location of the element of an array that matches an indicated value.

Copy of an array.

Reverse of an array.

```
import java.util.Scanner;
import java.util.Arrays;

class arrayy {
    public static void main(String[] args) {
        Scanner s1 = new Scanner(System.in);

        System.out.print("Enter size of array : ");
        int size = s1.nextInt();

        double[] arr = new double[size];

        System.out.println();

        for (int i = 0; i < size; i++) {
            System.out.print("Enter " + (i + 1) + " element : ");
            arr[i] = s1.nextDouble();
        }

        System.out.println();
        System.out.println("Length of array : " + arr.length);
        System.out.println();

        for (int i = 0; i < arr.length; i++) {
            System.out.println((i + 1) + " element : " + arr[i]);
        }

        double sum_sqr = 0;
        double sum_arr = 0;
        for (int i = 0; i < arr.length; i++) {
            sum_sqr += arr[i] * arr[i];
            sum_arr += arr[i];
        }

        System.out.println();
        System.out.println("Sum of square : " + sum_sqr);
```

```

System.out.println("Sum of array : " + sum_arr);
System.out.println();

double sum_arr_for_mean = 0;
for (int i = 0; i < arr.length; i++) {
    sum_arr_for_mean += arr[i];
}

System.out.println();
System.out.println("Mean of an array : " + (sum_arr_for_mean /
arr.length));
System.out.println();

Arrays.sort(arr);
double median;
if (arr.length % 2 != 0) {
    median = arr[arr.length / 2];
} else {
    median = (arr[(arr.length - 1) / 2] + arr[arr.length / 2])
/ 2.0;
}
System.out.println("Median : " + median);

System.out.println("\n1. for bubble sort (Ascending) \n2. for
selection sort (Descending)");
System.out.print("Enter choice : ");
int choice = s1.nextInt();

switch (choice) {
    case 1:
        int counter = 1;
        while (counter < arr.length) {
            for (int i = 0; i < arr.length - counter; i++) {
                if (arr[i] > arr[i + 1]) {
                    double temp = arr[i];
                    arr[i] = arr[i + 1];
                    arr[i + 1] = temp;
                }
            }
            counter++;
        }

        System.out.println("\nBubble sort (Ascending) : ");
        for (int i = 0; i < arr.length; i++) {
            System.out.println(arr[i] + " ");
        }
        System.out.println();
        break;

    case 2:
        for (int i = 0; i < arr.length - 1; i++) {

```

```

        for (int j = i + 1; j < arr.length; j++) {
            if (arr[j] > arr[i]) {
                double temp = arr[j];
                arr[j] = arr[i];
                arr[i] = temp;
            }
        }
    }

    System.out.println("\nSelection sort (Descending) : ");
    for (int i = 0; i < arr.length; i++) {
        System.out.println(arr[i] + " ");
    }
    System.out.println();
    break;

default:
    System.out.println("Invalid choice");
    break;
}

int additionalChoice;

do{
    System.out.println("\n1. Display array \n2. Display copied
array \n3. Display reversed array \n4. Search for an element \n5.
Exit");

    System.out.print("Enter choice : ");
    additionalChoice = sl.nextInt();

    switch (additionalChoice) {
        case 1:
            System.out.println("\nOriginal array:");
            for (int i = 0; i < arr.length; i++) {
                System.out.println(arr[i] + " ");
            }
            break;

        case 2:
            System.out.println("\nCopied array:");
            // Copy of an array
            double[] copied_array = new
double[arr.length];

            for (int i = 0; i < arr.length; i++) {
                copied_array[i] = arr[i];
            }
            System.out.println();

            // Display copied array

```

```

        System.out.println("Display copied array");
        for (int i = 0; i < arr.length; i++) {
            System.out.println(copied_array[i] +
" ");
        }

        break;

    case 3:
        System.out.println("\nReversed array:");
        for (int i = arr.length - 1; i >= 0; i--) {
            System.out.println(arr[i] + " ");
        }
        break;

    case 4:
        System.out.print("\nEnter Element to search : ");
        double value = s1.nextDouble();

        int foundIndex = -1;
        for (int i = 0; i < arr.length; i++) {
            if (arr[i] == value) {
                foundIndex = i;
                break;
            }
        }

        if (foundIndex != -1) {
            System.out.println("Element found at index: " +
foundIndex);
        } else {
            System.out.println("Element not found in the
array.");
        }
        break;

    case 5:
        System.out.println("Exiting.....");
        break;

    default:
        System.out.println("Invalid choice");
        break;
    }
}while(additionalChoice != 5 );

s1.close();
}
}

/* output

```

Enter size of array : 5

Enter 1 element : 78

Enter 2 element : 56

Enter 3 element : 94

Enter 4 element : 1

Enter 5 element : 25

Length of array : 5

1 element : 78.0

2 element : 56.0

3 element : 94.0

4 element : 1.0

5 element : 25.0

Sum of square : 18682.0

Sum of array : 254.0

Mean of an array : 50.8

Median : 56.0

1. for bubble sort (Ascending)

2. for selection sort (Descending)

Enter choice : 2

Selection sort (Descending) :

94.0

78.0

56.0

25.0

1.0

1. Display array

2. Display copied array

3. Display reversed array

4. Search for an element

5. Exit

Enter choice : 1

Original array:

94.0

78.0

56.0

25.0

1.0

1. Display array

2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

Enter choice : 2

Copied array:

Display copied array

94.0

78.0

56.0

25.0

1.0

1. Display array
2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

Enter choice : 3

Reversed array:

1.0

25.0

56.0

78.0

94.0

1. Display array
2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

Enter choice : 2

Copied array:

Display copied array

94.0

78.0

56.0

25.0

1.0

1. Display array
2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

Enter choice : 4

```
Enter Element to search : 95
Element not found in the array.
```

1. Display array
2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

```
Enter choice : 4
```

```
Enter Element to search : 94
Element found at index: 0
```

1. Display array
2. Display copied array
3. Display reversed array
4. Search for an element
5. Exit

```
Enter choice : 5
```

```
Exiting.....
```

```
*/
```

2) Write a program in Java to create variable size array. Data can be numeric.

Flush an array

add a number at a specified location in an array.

display the array

```
import java.util.Scanner;

class variable {
    private int[] array;
    private int size;

    public variable(int initial_size) {
        array = new int[initial_size];
        size = 0;
    }

    public void add_number(int number, int index) {

        if (index < 0 || index > size) {
            System.out.println("Invalid index. Please enter an index between
0 and " + (size + 1));
            return;
        }

        if (size == array.length) {
            resize_array();
        }
    }
}
```

```

    }

    for (int i = size; i > index; i--) {
        array[i] = array[i - 1];
    }

    array[index] = number;
    size++;
}

private void resize_array() {
    int new_capacity = array.length * 2;
    int[] new_array = new int[new_capacity];
    for (int i = 0; i < array.length; i++) {
        new_array[i] = array[i];
    }
    array = new_array;
}

public void flush_array() {
    array = new int[array.length];
    size = 0;
    System.out.println("Array has been flushed.");
}

public void display_array() {
    System.out.print("Array: ");
    for (int i = 0; i < size; i++) {
        System.out.print(array[i] + " ");
    }
    System.out.println();
}

public static void main(String[] args) {
    Scanner s1 = new Scanner(System.in);

    variable_array = new variable(2);

    System.out.println("Enter numbers to add to the array (type 'exit' to
stop):");
    while (s1.hasNextInt()) {
        int number = s1.nextInt();
        System.out.print("Enter the index to add the number: ");
        int index = s1.nextInt();
        variable_array.add_number(number, index);
        variable_array.display_array();
    }

    variable_array.flush_array();
    variable_array.display_array();

    s1.close();
}

```



```

}

/* output

Enter numbers to add to the array (type 'exit' to stop):

11
Enter the index to add the number: 0
Array: 11
12
Enter the index to add the number: 1
Array: 11 12
13
Enter the index to add the number: 2
Array: 11 12 13
14
Enter the index to add the number: 5
Invalid index. Please enter an index between 0 and 4
Array: 11 12 13
14
Enter the index to add the number: 3
Array: 11 12 13 14
15
Enter the index to add the number: 4
Array: 11 12 13 14 15
exit
Array has been flushed.
Array:

*/

```

3) Program in Java to find A+B, A-B, A*B and transpose of A, where A is a matrix of 3*3 and B is a matrix of 3*4. Take the values in matrixes A and B from the user.

```

import java.util.Scanner;

class matrix {

    public static void main(String[] args) {
        Scanner s1 = new Scanner(System.in);

        //declare
        int[][] A = new int[3][3];
        int[][] B = new int[3][4];

        // input matrix A
        System.out.println("Enter values for matrix A (3x3) :");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {

```

```

        A[i][j] = s1.nextInt();
    }
}

//display matrix A
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        System.out.print(A[i][j] + " ");
    }
    System.out.println();
}

// input matrix B
System.out.println("Enter values for matrix B (3x4) :");
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        B[i][j] = s1.nextInt();
    }
}

//display matrix B
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        System.out.print(B[i][j] + " ");
    }
    System.out.println();
}

// multiplication of A and B
int[][] mul = new int[3][4];
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        mul[i][j] = 0;
        for (int k = 0; k < 3; k++) {
            mul[i][j] += A[i][k] * B[k][j];
        }
    }
}

//display multiplication
System.out.println("Product of A and B matrices is : ");
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        System.out.print(mul[i][j] + " ");
    }
    System.out.println();
}

//original matrix display
System.out.println("Original Matrix : ");
for (int i = 0; i < 3; i++) {

```

```

        for (int j = 0; j < 3; j++) {
            System.out.print(A[i][j] + " ");
        }

        System.out.println();
    }

    //transpose logic
    int[][] transpose = new int[3][3];
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            transpose[i][j] = A[j][i];
        }
    }

    //display transpose
    System.out.println("Transpose Matrix : ");
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            System.out.print(transpose[i][j] + " ");
        }
        System.out.println();
    }
}

```

/* output

Enter values for matrix A (3x3) :

```

12
78
45
79
36
25
14
25
14
12 78 45
79 36 25
14 25 14

```

Enter values for matrix B (3x4) :

```

1
1
1
2
2
2
3
3
3
3

```

```
4
4
4
1 1 1 2
2 2 3 3
3 4 4 4
```

```
Product of A and B matrices is :
303 348 426 438
226 251 287 366
106 120 145 159
```

```
Original Matrix :
12 78 45
79 36 25
14 25 14
```

```
Transpose Matrix :
12 79 14
78 36 25
45 25 14
```

```
*/
```

4) Write and run a JAVA program that reads a string from the user and perform the following.

counts number of occurrence of a given character (for example, " a") in a string.

searches the last occurrence of a character in a string.

removes the unnecessary spaces from a string : leading and trailing spaces.

displays the substring formed by the last ten characters of a string

```
import java.util.Scanner;
```

```
class opr {
```

```
    public static void main(String[] args) {
        Scanner s1 = new Scanner(System.in);
```

```
        //input string from the user
        System.out.println("Enter a string : ");
        String input_string = s1.nextLine();
```

```
        // character to search for its occurrences and last position
```

```

        System.out.println("Enter a character to search : ");
        char search_char = s1.next().charAt(0);

        // count the number of occurrences
        int occurrence_count = count_occurrences(input_string,
search_char);
        System.out.println("Number of occurrences of '" + search_char +
"' : " + occurrence_count);

        // find the last occurrence of the given character
        int last_index = input_string.lastIndexOf(search_char);
        if (last_index != -1) {
            System.out.println("Last occurrence of '" + search_char +
"' : Index " + last_index);
        }
        else {
            System.out.println("Character '" + search_char + "' not
found in the string.");
        }

        // Remove leading and trailing spaces
        String trimmed_string = input_string.trim();
        System.out.println("String after removing leading and trailing
spaces: '" + trimmed_string + "'");

        // Display the substring formed by the last ten characters of
the string
        String last_ten_chars =
get_last_ten_characters(trimmed_string);
        System.out.println("Last ten characters of the string: '" +
last_ten_chars + "'");

        // Close the scanner
        s1.close();
    }

    // Method to count the number of occurrences of a given character
in a string
    public static int count_occurrences(String str, char ch) {
        int count = 0;
        for (int i = 0; i < str.length(); i++) {
            if (str.charAt(i) == ch) {
                count++;
            }
        }
        return count;
    }

    // Method to get the last ten characters of a string
    public static String get_last_ten_characters(String str) {

```

```

        if (str.length() <= 10) {
            return str;
        }
        else {
            return str.substring(str.length() - 10);
        }
    }
}

/* output

Enter a string :
hello how are you
Enter a character to search :
w
Number of occurrences of 'w': 1
Last occurrence of 'w': Index 8
String after removing leading and trailing spaces: 'hello how are you'
Last ten characters of the string: 'ow are you'

*/

```

5) WAP that inputs a line of text, tokenizes the line with StringTokenizer and outputs the tokens in reverse order.

```

import java.util.StringTokenizer;
import java.util.Scanner;

class reverse_tokenize {
    public static void main(String[] args) {
        Scanner s1 = new Scanner(System.in);

        System.out.println("Enter a line of text:");
        String input = s1.nextLine();

        StringTokenizer tokenizer = new StringTokenizer(input);
        String[] tokens = new String[tokenizer.countTokens()];

        int index = 0;
        while (tokenizer.hasMoreTokens()) {
            tokens[index] = tokenizer.nextToken();
            index++;
        }

        for (int i = tokens.length - 1; i >= 0; i--) {
            System.out.println(tokens[i]);
        }

        s1.close();
    }
}

```

```
/* output

Enter a line of text:
hello how are you

you
are
how
hello
*/
```

6) Create a StringBuffer and illustrate how to append character. Display capacity, length of the StringBuffer.

```
import java.util.Scanner;

class Main {
    public static void main(String[] args) {
        Scanner s1= new Scanner(System.in);

        System.out.print("Enter a character to append: ");
        char character = s1.next().charAt(0);

        StringBuffer sb= new StringBuffer();
        sb.append(character);

        System.out.println("String Buffer Capacity: " + sb.capacity());
        System.out.println("String Buffer Length: " + sb.length());
    }
}

/* output

Enter a character to append: K
String Buffer Capacity: 16
String Buffer Length: 1

*/
```

7) Write an application that reads and processes strings from the console. Perform the following functions based on the menu choice selected by the user..

- Reverse the sequence of strings and then display it.**
- Reverse the sequence of characters in each string and then display it.**
- rearrange the strings according to the length of the string.**

Sorting

Concatenation

Change them to uppercase, lowercase depending on user's choice.

```
import java.util.Scanner;

class string_processor {

    private static final int MAX_STRINGS = 100;
    private static String[] strings = new String[MAX_STRINGS];
    private static int count = 0;

    public static void main(String[] args) {
        Scanner s1 = new Scanner(System.in);
        boolean running = true;

        while (running) {

            System.out.println("\nMenu:");
            System.out.println("1. Enter strings");
            System.out.println("2. Reverse the sequence of strings");
            System.out.println("3. Reverse characters in each string");
            System.out.println("4. Rearrange strings by length");
            System.out.println("5. Sort strings alphabetically");
            System.out.println("6. Concatenate strings");
            System.out.println("7. Change case of strings");
            System.out.println("8. Exit");

            System.out.print("Enter your choice (1-8): ");
            int choice = s1.nextInt();
            s1.nextLine();

            switch (choice) {
                case 1:

                    System.out.println("Enter strings (type 'done' when
finished):");

                    count = 0;
                    String input;
                    while (count < MAX_STRINGS && !(input =
s1.nextLine()).equalsIgnoreCase("done")) {
                        strings[count++] = input;
                    }
                    break;

                case 2:

                    reverse_strings();
                    System.out.println("Reversed sequence of strings:");
                    print_strings();
                    break;
```


case 3:

```
        System.out.println("Strings with characters reversed:");
        for (int i = 0; i < count; i++) {
            String reversed = new
StringBuilder(strings[i]).reverse().toString();
            System.out.println(reversed);
        }
        break;
```

case 4:

```
        sort_by_length();
        System.out.println("Strings sorted by length:");
        print_strings();
        break;
```

case 5:

```
        sort_alphabetically();
        System.out.println("Strings sorted alphabetically:");
        print_strings();
        break;
```

case 6:

```
        String concatenated = concatenate_strings();
        System.out.println("Concatenated string:");
        System.out.println(concatenated);
        break;
```

case 7:

```
        System.out.println("Choose case transformation:");
        System.out.println("1. Uppercase");
        System.out.println("2. Lowercase");
        int caseChoice = s1.nextInt();
        s1.nextLine();

        if (caseChoice == 1) {
            to_upper_case();
            System.out.println("Strings in uppercase:");
        }
        else if (caseChoice == 2) {
            to_lower_case();
            System.out.println("Strings in lowercase:");
        }
        else {
            System.out.println("Invalid choice.");
        }
        print_strings();
        break;
```

```

        case 8:

            running = false;
            System.out.println("Exiting...");
            break;

        default:
            System.out.println("Invalid choice. Please select a
number between 1 and 8.");
    }
}

s1.close();
}

private static void reverse_strings() {
    for (int i = 0; i < count / 2; i++) {
        String temp = strings[i];
        strings[i] = strings[count - 1 - i];
        strings[count - 1 - i] = temp;
    }
}

private static void sort_by_length() {
    for (int i = 0; i < count - 1; i++) {
        for (int j = i + 1; j < count; j++) {
            if (strings[i].length() > strings[j].length()) {
                String temp = strings[i];
                strings[i] = strings[j];
                strings[j] = temp;
            }
        }
    }
}

private static void sort_alphabetically() {
    for (int i = 0; i < count - 1; i++) {
        for (int j = i + 1; j < count; j++) {
            if (strings[i].compareTo(strings[j]) > 0) {
                String temp = strings[i];
                strings[i] = strings[j];
                strings[j] = temp;
            }
        }
    }
}

private static String concatenate_strings() {
    StringBuilder sb = new StringBuilder();
    for (int i = 0; i < count; i++) {
        sb.append(strings[i]).append(" ");
    }
    return sb.toString().trim();
}

```

```

    }

    private static void to_upper_case() {
        for (int i = 0; i < count; i++) {
            strings[i] = strings[i].toUpperCase();
        }
    }

    private static void to_lower_case() {
        for (int i = 0; i < count; i++) {
            strings[i] = strings[i].toLowerCase();
        }
    }

    private static void print_strings() {
        for (int i = 0; i < count; i++) {
            System.out.println(strings[i]);
        }
    }
}

/* output

```

D:\>java string_processor

Menu:

1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 1

Enter strings (type 'done' when finished):

hello how r u

this is a car

string buffer

subject operating system

done

Menu:

1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 2
Reversed sequence of strings:
subject operating system
string buffer
this is a car
hello how r u

Menu:
1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 3
Strings with characters reversed:
metsys gnitarepo tcejbus
reffub gnirts
rac a si siht
u r woh olleh

Menu:
1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 4
Strings sorted by length:
string buffer
this is a car
hello how r u
subject operating system

Menu:
1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 5
Strings sorted alphabetically:

hello how r u
string buffer
subject operating system
this is a car

Menu:

1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 6

Concatenated string:

hello how r u string buffer subject operating system this is a car

Menu:

1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 7

Choose case transformation:

1. Uppercase
2. Lowercase

1

Strings in uppercase:

HELLO HOW R U

STRING BUFFER

SUBJECT OPERATING SYSTEM

THIS IS A CAR

Menu:

1. Enter strings
2. Reverse the sequence of strings
3. Reverse characters in each string
4. Rearrange strings by length
5. Sort strings alphabetically
6. Concatenate strings
7. Change case of strings
8. Exit

Enter your choice (1-8): 7

Choose case transformation:

1. Uppercase

2. Lowercase

2

Strings in lowercase:

hello how r u

string buffer

subject operating system

this is a car

Menu:

1. Enter strings

2. Reverse the sequence of strings

3. Reverse characters in each string

4. Rearrange strings by length

5. Sort strings alphabetically

6. Concatenate strings

7. Change case of strings

8. Exit

Enter your choice (1-8): 8

Exiting...

*/