



---

# RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOGY

---

Department of Computer Science & Engineering

## LAB REPORT

Topic: Shell Script (Array)

Course No: CSE 3202

Course Name: Sessional Based on Operating Systems

Submitted By

Saifur Rahman  
Roll No: 1703018  
Section: A  
CSE, RUET

Submitted To

Mohiuddin Ahmed  
Lecturer  
CSE, RUET

Date of Lab: 4<sup>th</sup> Apr 2022

Date of Submission: 16<sup>th</sup> May 2022

## Contents

Program Topic: Find the maximum and minimum number of an array.....	1
Program Topic: Sort the elements of an array in ascending and descending order. ....	2
Discussion: .....	3

## Program No. 1

Program Topic: Find the maximum and minimum number of an array.

### CODE

```
#!/bin/bash

read -a arr
echo Value of Arrays: ${arr[@]}

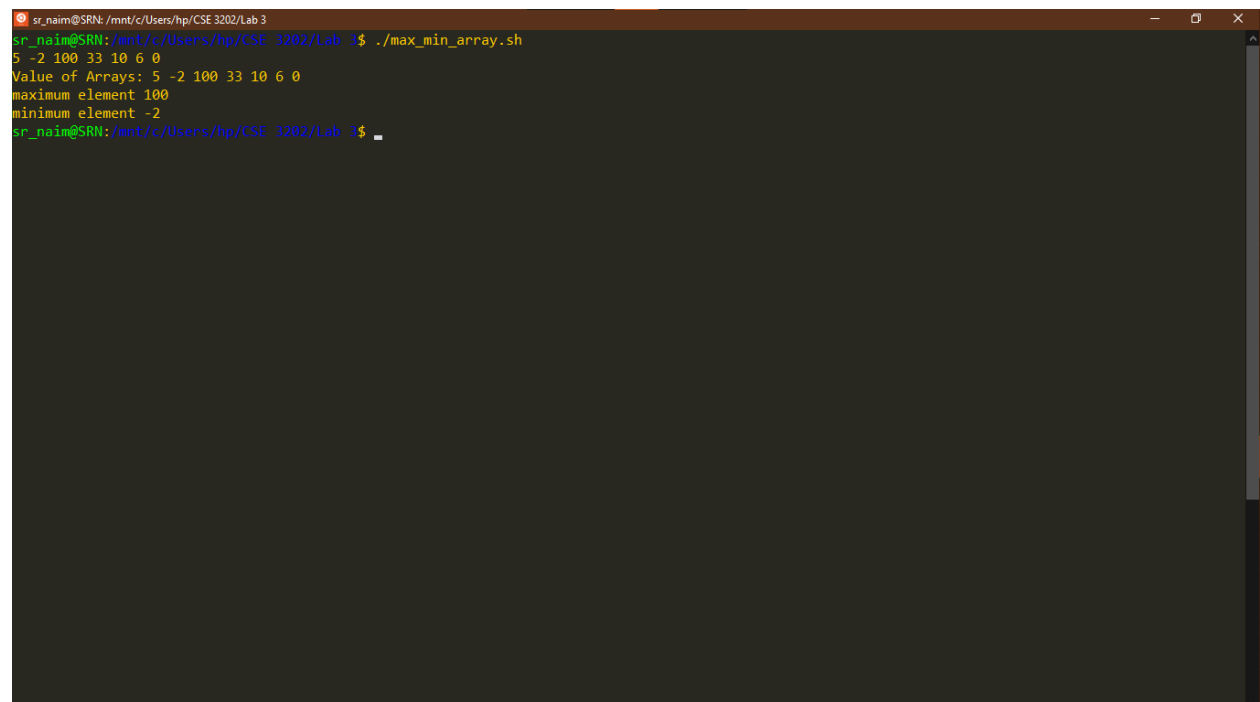
max=${arr[0]}
min=${arr[0]}

for i in ${arr[@]}
do
    if (( $i >= $max ))
    then
        max=$i
    fi

    if (( $i <= $min ))
    then
        min=$i
    fi
done

echo maximum element $max
echo minimum element $min
```

### OUTPUT



```
sr_naim@SRN: /mnt/c/Users/hp/CSE 3202/Lab 3
sr_naim@SRN: /mnt/c/Users/hp/CSE 3202/Lab 3$ ./max_min_array.sh
5 -2 100 33 10 6 0
Value of Arrays: 5 -2 100 33 10 6 0
maximum element 100
minimum element -2
sr_naim@SRN: /mnt/c/Users/hp/CSE 3202/Lab 3$ _
```

## Program No. 2

Program Topic: Sort the elements of an array in ascending and descending order.

### CODE

```
#!/bin/bash
```

```
read -a arr
```

```
echo Values of Arrays: ${arr[@]}
```

```
len=${#arr[@]}
```

```
echo Length: $len
```

```
# bubble sort
```

```
temp=0
```

```
# Ascending
```

```
for ((i=0;i<${len-1};i++))
```

```
do
```

```
    for ((j=0;j<${len-i-1};j++))
```

```
    do
```

```
        if (( ${arr[j]} > ${arr[${j+1}]} ))
```

```
        then
```

```
            temp=${arr[j]}
```

```
            arr[j]=${arr[${j+1}]}
```

```
            arr[${j+1}]=$temp
```

```
        fi
```

```
    done
```

```
done
```

```
echo Array in ascending order: ${arr[@]}
```

```
# Descending
```

```
for ((i=0;i<${len-1};i++))
```

```
do
```

```
    for ((j=0;j<${len-i-1};j++))
```

```
    do
```

```
        if (( ${arr[j]} < ${arr[${j+1}]} ))
```

```
        then
```

```
            temp=${arr[j]}
```

```
            arr[j]=${arr[${j+1}]}
```

```
            arr[${j+1}]=$temp
```

```
        fi
```

```
    done
```

```
done
```

```
echo Array in descending order: ${arr[@]}
```

## OUTPUT

```

sr_naim@SRN: /mnt/c/Users/hp/CSE 3202/Lab 3
sr_naim@SRN:/mnt/c/Users/hp/CSE 3202/Lab 3$ ls -al
total 4
drwxr-xr-x 1 sr_naim sr_naim 4096 May 16 06:31 .
drwxrwxrwx 1 sr_naim sr_naim 4096 May 15 23:23 ..
-rwxr--r-- 1 sr_naim sr_naim 626 Apr  4 12:33 array_sorting.sh
-rwxr--r-- 1 sr_naim sr_naim 125 Apr  4 11:27 conditional_statements.sh
-rwxr--r-- 1 sr_naim sr_naim 249 Apr  4 12:05 max_min_array.sh
sr_naim@SRN:/mnt/c/Users/hp/CSE 3202/Lab 3$ vim array_sorting.sh
sr_naim@SRN:/mnt/c/Users/hp/CSE 3202/Lab 3$ ./array_sorting.sh
3 -2 7 0 51 8 10 13 30 -6 8
Values of Arrays: 3 -2 7 0 51 8 10 13 30 -6 8
Length: 11
Array in ascending order: -6 -2 0 3 7 8 8 10 13 30 51
Array in descending order: 51 30 13 10 8 8 7 3 0 -2 -6
sr_naim@SRN:/mnt/c/Users/hp/CSE 3202/Lab 3$

```

## Discussion:

- In shell scripting, when we access the value of a variable, we use the '\$' sign. Otherwise, we do not need to use the dollar sign. For example, while assigning a value to a variable we do not use the dollar sign for the variable we are going to assign on.
- To print all the elements of array an '\*' sign can also be used other than '@' sign. We can print all the elements in this way too E.g. '\${arr[@]:0}'.
- Only the array name without any index specified denotes the first element E.g. '\${arr}'.
- While using conditional statements or array indexes, we have to be precautious about first brackets and extra spaces. Both could cause an error in the code.