



Week 1: Assignment 4

Basic Trigonometry with Shell Scripting

[Last Updated on: 08th March 2021, 16:00 Hrs]

- Aim
- Given
- Procedure
- Expected Output
- Grading and Submission Instructions
- References

Aim

In this assignment, you will be converting the given Cartesian x and y coordinates to Polar coordinates r and θ . You will be using few shell commands to work with input arguments in a shell file. The x and y coordinates will be provided as input arguments to the shell script.

The program is ideally expected to print the r and θ coordinates corresponding to the input x and y coordinates upto five decimal places. The θ value can have range from 0 to 360 degrees.

Given

One file is provided to solve this assignment.

- Skeleton program file: `assignment4.sh`

Procedure

- Open the skeleton program file, `assignment4.sh`.
- You will notice pre-written comments included in skeleton program for your assistance to solve the assignment.
- To run and debug your solution, type the below command in Terminal:

```
$ bash assignment4.sh 10 10
```



This command will run the shell script `assignment4.sh` with the x and y coordinates as input arguments.

- Refer the **Expected Output** section below and debug your code to get the correct output.

Expected Output

- For example, the input arguments of Cartesian coordinates are $x = 10$ and $y = 10$.
- The expected output of program **assignment4.sh** i.e., print the corresponding r and θ values upto 5 decimal places is shown below:

```
14.14214, 45.00000
```



- You can test your solution with following values and their expected outputs:

```
$ bash assignment4.sh -10 10
```

```
14.14214, 135.00000
```



```
$ bash assignment4.sh -10 -10
```

```
14.14214, 225.00000
```

```
$ bash assignment4.sh 10 -10
```

```
14.14214, 315.00000
```

Grading and Submission Instructions

- Navigate to the folder where the **ey-mooc-grader-sfc** application resides.
- To grade your solution, run the **check** command of the application as follows:

```
$ ./ey-mooc-grader-sfc check -w 1 -a 4 Week_1/Assignment_4/assignment4.sh
```



- This will run your program **assignment4.sh** against random test cases and grade it. Marks and appropriate remarks will be provided as shown in Figure 1.
- Your program file **assignment4.sh**, marks scored and remarks will get uploaded to the MOOC portal.

```

erts-09@erts:~/Desktop/SFC_PartI_MOOC
File Edit View Search Terminal Help
~/Desktop/SFC_PartI_MOOC 20:20:20
./ey-mooc-grader-sfc check -w 1 -a 4 Week_1/Assignment_4/assignment4.sh

Course Name: Software Foundation (Part I)

Checking your submission for Week - 1 Assignment number - 4

Checking submission type ...
Submission type is accepted

Downloading test scripts ...
100% [.....] 4630 / 4630
Download complete

Extracting files ...
Extraction complete

#### RESULT ####
+-----+-----+-----+-----+
| TEST CASE NUMBER | TEST CASE PASSED? (Y/N) | MARKS SCORED | REMARKS |
+-----+-----+-----+-----+
| 1 | Y | 2.0 | Good work. |
| 2 | Y | 4.0 | Good work. |
| 3 | Y | 6.0 | Good work. |
| 4 | Y | 8.0 | Good work. |
| 5 | Y | 10.0 | Good work. |
+-----+-----+-----+-----+

REMARKS = Congrats! You have successfully completed the assignment. Keep it up!
MARKS = 10

MARKS AND REMARKS UPLOADED ON THE PORTAL SUCCESSFULLY
~/Desktop/SFC_PartI_MOOC 20:20:32

```

Figure 1: Output of running check command for Week 1 Assignment 4

- You can verify this by running the **status** command of the application as given below, refer Figure 2.

```
$ ./ey-mooc-grader-sfc status -w 1 -a 4
```



```
erts-09@erts:~/Desktop/SFC_PartI_MOOC
File Edit View Search Terminal Help
~/Desktop/SFC_PartI_MOOC 20:20:55
$ ./ey-mooc-grader-sfc status -w 1 -a 4
Course Name: Software Foundation (Part I)
Checking status of your submission for Week - 1 Assignment number - 4
#### LAST RECORDED RESULT ####
REMARKS      : Congrats! You have successfully completed the assignment. Keep it up!
MARKS        : 10
UPLOAD DATE-TIME : 2021-04-03 20:20:32
~/Desktop/SFC_PartI_MOOC 20:20:59
```

Figure 2: Output of running status command for Week 1 Assignment 4

References

- Nano Editor
 - [How to use Nano Text Editor](#)
 - [Nano Editor Official Docs](#)
- Vim Editor
 - [Interactive Vim Tutorial](#)
- Passing arguments to Bash script
 - [Bash Scripting Tutorial](#)
- Bash Arithmetic
 - [Bash Arithmetic Operations](#)
 - [Basic Calculator \(bc\) command manual](#)
 - [bc command examples](#)
 - [Arithmetic Comparisons](#)
- Bash Conditional Statement
 - [Syntax and use of Conditional Statments in Bash](#)
 - [Shell Scripting Tutorial on Conditional Statements](#)

All The Best!