**IS 340 – Operating Systems**

**HOP10 – BASH – String Manipulation**

7/18/2019 Developed by Kevin Wang

7/20/2020 Reviewed by Kim Nguyen

02/14/2022 Reviewed by Ken Ling

School of Technology & Computing (STC) @ City University of Seattle (CityU)

**Before You Start**

* This exercise assumes that the user is working with the Ubuntu 18.04 distribution. If you are working with a different Linux distribution, the set of shell commands may vary from those available in Ubuntu 18.04.
* Students will use the EC2 Ubuntu virtual machine that they created in the module 1 exercise.
* All commands and code discussed in this exercise will run in the Ubuntu console.
* The directory path shown in screenshots may be different from yours.
* Some steps are not explained in the tutorial**.** If you are not sure what to do:
  1. Consult the resources listed below and experiment in the Ubuntu console and try to solve the problem yourself.
  2. If you cannot solve the problem after a few tries, ask a TA for help.

**Learning Outcomes**

Students will be able to:

* Concatenate strings
* Process character by character and reverse a string
* Convert cases
* Insert characters into a string
* Trim unwanted characters

**Resources**

# Linux command line: bash + utilities

<https://ss64.com/bash/>

* Nano/Basics Guide

<https://wiki.gentoo.org/wiki/Nano/Basics_Guide>

**Preparation**

1. Connect to your Ubuntu instance

Open a command prompt

Syntax: ssh -i LOCATION\_OF\_YOUR\_KEY ubuntu@PULIC\_DNS

Example:

>>>ssh -i key.pem ubuntu@ec2-33-222-101-222.us-west-2.compute.amazonaws.com

1. Navigate to your name directory under the IS340-Summer-2020

>>> cd ~/ IS340-Summer-2020/YOURNAME

Note: change YOURNAME to your real name

1. Create a Module9 directory under YOURNAME directory.

Note: If this directory exists, skip this step.

>>> mkdir Module9

1. Navigate to the Module9 directory.

>>> cd Module9

**Concatenate strings**

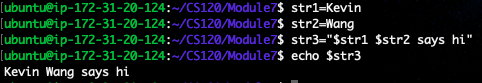
1. Concatenating strings is very easy in the Bash. Try it by typing follow commands:

>>> str1=Kevin

>>> str2=Wang

>>> str3="$str1 $str2 says hi"

>>> echo $str3



1. After Bash 3.1, we can use += operator to concatenate strings. Test it by typing the following command:

>>> str3+=” there.”

>>> echo $str3

Note: directly copy commands will not work since the Word uses different double quote marks. Please type the command by yourself.



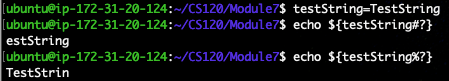
**Process character by character and reverse a string**

1. Test the following commands to see how bash script can get whole string without the first or last character:

>>> testString=TestString

>>> echo ${testString#?}

>>> echo ${testString%?}

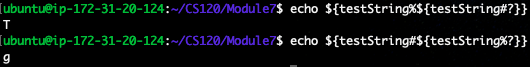


Note: #? means marking one character from beginning and %? means marking one character from the end. You can use #?? to mark two characters.

1. So, we can use the follow combined commands to get the first or last character:

>>> echo ${testString%${testString#?}}

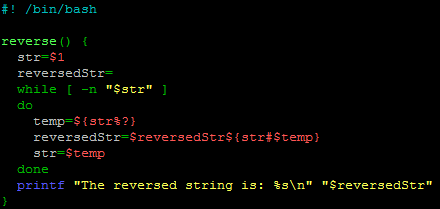
>>> echo ${testString#${testString%?}}



1. Try to use the technic we learned above to make a function that helps to reverse a string. Type the following command to create a script file:

>>> nano Reverse.sh

1. Type the script into the file as below:



1. Hit the control + x to quit and save the file
2. Type the following command to source the file:

>>> . Reverse.sh

1. Type the following command to test the reverse function:

>>> reverse abcde



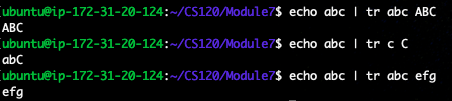
**Case conversion**

1. The tr command can help to make a character map. Test it by typing the following commands:

>>> echo abc | tr abc ABC

>>> echo abc | tr c C

>>> echo abc | tr abc efg



Note: as you can see the tr command is offering a map that can map a character to any other character.

1. Try this command to map a group of characters:

>>> echo “This is a full sentence.” | tr ‘a-z’ ‘A-Z’



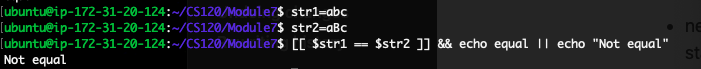
Note: please do not directly copy the command.

1. Sometimes, we want to compare two strings that have the same characters with different cases. We can convert both strings to uppercase and compare them. Try it by typing the following commands:

>>> str1=abc

>>> str2=aBc

>>> [[ $str1 == $str2 ]] && echo equal || echo "Not equal"



The result shows they are not equal since we have different cases for the letter b.

Use the following commands to overcome this:

>>> [[ ${str1^^} == ${str2^^} ]] && echo equal || echo "Not equal"



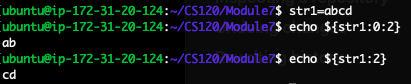
**Insert characters in a string**

1. In order to perform the insertion, we have to separate the original string to the left and right part. Try this by typing the following commands:

>>> str1=abcd

>>> echo ${str1:0:2}

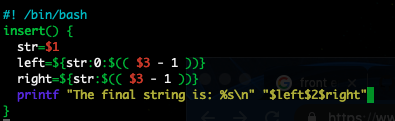
>>> echo ${str1:2}



1. Now we can use this approach to make an insert function. Create a script file by typing the following command:

>>> nano Insert.sh

1. Type the following script in the file:



1. Hit the control + x key to quit and save the file:
2. Test the following commands to test the new function:

>>> . Insert.sh

>>> insert Hllo e 2

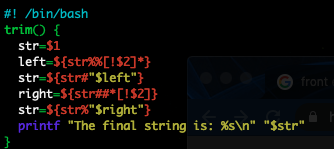


**Trim unwanted characters**

1. Type the following command to create a trim script:

>>> nano Trim.sh

1. Type the following script in the file:



1. Hit the control + x key to quit and save the file.
2. Test the script by typing follow commands:

>>> . Trim.sh

>>> trim 0000213.43000 0



**Push your work to GitHub**

Run the following commands to push your work to the GitHub repository:

>>> git add .

>>> git commit -m “Submission for Module 10”

>>> git push origin YOUR\_BRANCH\_NAME

Note: you should change the YOUR\_BRANCH\_NAME to your own branch name. It should be firstname-lastname (e.g. maria-gracia).

If you cannot remember, run the command “git status” to check.