

Operating System Lab Final Exam

Time : 35 min ; Marks : 25

1. Write a shell script program named **passwordChecker.sh**, which will identify the flaws of a password based on the condition given below.
 - i. If the length of the password is less than equal 10 : it will print *"Password needs to more than 10 characters"*
 - ii. If the The password doesn't contain any numbers then it will print *"Password must contain at least one number"*
 - iii. If the password does not contain any special characters then it will print *"Password must contain a special character."*

If all the conditions mentioned are properly satisfied then it will print "All good" and append the password in a file named **StrongPassword.txt**.

15

Input	Output
akiz	Password needs to more than 10 characters
akizwithten	Password must contain at least one number
akizwith10	Passwords must contain a special character.
akizwith10@	All good <i>[appends the password in a file named StrongPassword.txt]</i>

2. Create a shell script program named sumOfdigits.sh, where you will pass your last 3 digits of your id as an **argument** and it will calculate the sum of the last 3 digits of your id and print it in the terminal. 10

Input	Output
213	6
920	11

grep command extension

<u>grep Option</u>	<u>Meaning</u>	<u>Example</u>	<u>Output</u>
	Not in "[^pattern]"	<code>grep "[^0-9]" file.txt</code>	<u>Matches lines containing non-digit characters</u>
<u>-i</u>	<u>Ignore case</u>	<code>grep -i "hello" file.txt</code>	<u>matches: Hello, HELLO, hello</u>
<u>-v</u>	<u>Invert match</u>	<code>grep -v "error" log.txt</code>	<u>all lines without "error"</u>
<u>-c</u>	<u>Count</u>	<code>grep -c "fail" log.txt</code>	<u>3 (if 3 lines match)</u>
<u>-l</u>	<u>File names only</u>	<code>grep -l "error" *.log</code>	<u>app.log</u>
<u>-w</u>	<u>Whole word</u>	<code>grep -w "cat" file.txt</code>	<u>matches cat but not catch</u>
<u>-x</u>	<u>Whole line</u>	<code>grep -x "hello" file.txt</code>	<u>hello (exact line only)</u>
<u>-e</u>	<u>Multiple patterns</u>	<code>grep -e "error" -e "fail" log.txt</code>	<u>lines containing error OR fail</u>
<u>-o</u>	<u>Only match</u>	<code>grep -o "[0-9]" file.txt</code>	<u>1 2 3 (each digit separately)</u>

Length calculation	<code>name="HelloWorld"</code> <code>len=\${#name}</code>
--------------------	--

condition + loop + pipeline syntax

<code>echo "Enter a number:" read num if [\$num -eq 10] then echo "Number is 10" elif [\$num -gt 10] then echo "Number is greater than 10" else echo "Number is less than 10" fi</code>	<code>for ((i=1; i<=5; i++)) do echo \$i done a=10 b=5 echo \$((a + b)) echo \$((a - b)) echo \$((a * b)) echo \$((a / b))</code>	<code>while [condition] do commands done cat file.txt grep "hello" #Shows lines containing "hello" cat file.txt grep "error" wc -l #Counts how many times "error" appears</code>
--	--	--

Permission	chmod [permissions] filename	
	chmod u=rwx,g=rx,o=r file.txt	chmod u+x file.sh