Shell Scripting

Multiple Choice And Theory Question

1] How can you find out how long the system has been running?

a. Command “uptime”

b. Command “time”
c. Command “datetime”
d. None

2] How to get input from the terminal for shell script?

a. 'input' command
b. 'read' command
c. 'echo' command
d. None

3] What is the use of “$?” sign in shell script?

a. Print the name of the shell.
b. No. of arguments to a shell script.
c. Check whether previous command is executed successfully or not
d. None

4] What are the redirect options to use for sending both standard output and standard error to the same location?

a. 2 >&
b. &>
c. 2>&1
d. Both b and c

5] How to display all array indexes at once?

a. echo ${!array[@]}
b. echo ${array[*]}
c. echo ${array[@]}
d. None

6] What is the difference between $$ and $!?

a. $$ gives the last error code of the currently executing process whereas $! returns the exit code of the process that recently went into background.
b. $$ gives the no. of arguments of the currently executing process whereas $! holds the list of arguments of the process that recently went into background.
c. $$ gives the process id of the currently executing process whereas $! shows the process id of the process that recently went into background.
d. None

7] Given a file find the count of lines containing word "ABC"

a. grep c- "ABC" file1
b. grep -c "ABC" fle1
c. grep c "ABC" file1
d. None

8] How to remove array element with id 3?

a. remove array[2]
b. unset array[2]
c. remove array[1]
d. unset array[3]

9] How to add new array element with id 99?

a. array[98]="New_element"
b. set array[99]="New_element"
c. array[99]="New_element"
d. None

10] How can you set the default rwx permission to all users on every file which is created in the current shell?

a. umask 555
b. umask 666
c. umask 777
d. Umask 888

11] What is the use of $# in shell scripting?

a. Exit code of the shell script.
b. Error code of the command last executed.
c. Count of the arguments passed to a shell script
d. None

12] How to display the first element of an array?

a. echo array[1]
b. echo ${array[1]}
c. echo ${array[0]}
d. echo array[0]

13] How to define array in shell script?

a. array=["Hello" "We" "are" "TechBeamers"]
b. array=("Hello" "We" "are" "TechBeamers")
c. array="Hello","We" "are","TechBeamers"
d. array={"Hello" "We" "are" "TechBeamers"}

14] How to debug a shell script?

a. sh -x testscript.sh
b. sh -d testscript.sh
c. sh -nv testscript.sh
d. Both a and c

15] What is the correct comparison statement in Linux shell Scripting?

a. if ($x -gt $y)
b. if [$x -gt $y]
c. if $x -gt $y
d. None

Theory Question

1] Write a program to find prime number between 1 to 100 using shell scripting

2] Write a calculator program using switch case [Addition , Substraction , division ,and Multiplication].

3] Write a program to find the given number is palindrome or not using shell scripting.

4] Write a program to sort a array in asending order using shell scripting.

5] Write a program to find the number of words in a string using shell scripting.

6] Wite a program to reverse a array elements using shell scripting.

7] Write a program to delete repeating elements in array.

8] What is Bash ? What is chmod ? What are all possible modes ? How to chnage the mode ?

9] Write a program to reverse a string using shell scripting

10] Write a program to find factorial of given number using shell scripting

Solution

1] Write a program to find prime number between 1 to 100 using shell scripting

```
#!/bin/bash

echo "Enter the min Start of range:"
read -p "Min=>" min

echo "Enetr the max range:"
read -p "Max=>" max

for((index1=1;index1<=$max;index1++))
do
	flag=0

	for((index2=2;index2<=`expr $i / 2`;index2++))
	do
		if [ `expr $index1 % $index2` -eq 0 ]
		then
			flag=1
		fi
	done

	if [ $flag -eq 0 ]
	then
		echo  "$index1"
	fi

done
```

2] Write a calculator program using switch case [Addition , Substraction , division ,and Multiplication].

```
#!/bin/bash

echo "Enetr the two number to do mathematical operation:"
read -p "Number1=>" number1
read -p "Number2=>" number2

echo "Enetr the choice 1] Add 2]Sub 3]mul 4]Div "
read -p "Choice=>" choice

case "$choice" in

        Add | add) echo "Addtion = `expr $number1 + $number2`";;
        Sub | sub) echo "Substration= `expr $number1 - $number2`";;
        Mul| mul) echo " Multiplication= `expr $number1 \* $number2`";;
        Div | div) echo "Division= `expr $number1 / $number2`";;
        *) echo "Answer not recognized";;
esac
```

3] Write a program to find the given number is palindrome or not using shell scripting.

```
#!/bin/bash

echo "Enter a number:"
read -p "Number=>" number

temp=$number
reverse=0

while [ $temp -ne 0 ]
do
        modulus=`expr $temp % 10`
        mul=`expr $reverse \* 10`
        reverse=`expr $mul + $modulus`
        temp=`expr $temp / 10`
done

if [ $number -eq $reverse ]
then
        echo "Palindrome"
else
        echo "Not a palindrome"
fi
```

4] Write a program to sort a array in asending order using shell scripting.

```
#!/bin/bash

echo "Enetr the length of array:"
read -p "length=>" length

for(( index1=0;index1< $length;index1++))
do
        echo "Enetr `expr $index1 + 1`"
        read array[$index1]
done

for(( index1=0;index1< $length;index1++))
do
        for(( index2=`expr $index2 + 1`;index2< $length;index2++))
        do
                if [ ${array[$index1]} -gt ${array[$index2]} ]
                then
                        temp=${array[$index1]}
                        array[$index1]=${array[$index2]}
                        array[$index2]=$temp
                fi
        done
done

echo "Array asendig order :"

for(( index1=0;index1< $length;index1++))
do
        echo array[$index1] = ${array[$index1]}
done
```

5] Write a program to find the number of words in a string using shell scripting.

```
#/bin/bash

echo "Enetr the input string:"
read -p "string=>" string

echo $string | wc -w
```

6] Wite a program to reverse a array elements using shell scripting.

```
#/bin/bash

echo "Enter the array length:"
read -p "length=>" length

echo "Enetr the array element:"
for (( index1=0;index1< $length;index1++ ))
do
        echo "Enter `expr $index1 + 1`:"
        read array[$index1]
done
for(( index1=0, index2=`expr $length - 1` ; index2 > index1;index1++,index2-- ))
do
        temp=${array[$index1]}
        array[$index1]=${array[$index2]}
        array[$index2]=$temp
done

echo "reverse array element:"
for((index1=0;index1< $length ; index1++))
do
        echo "array[$index1]=${array[$index1]}"
done
```

7] Write a program to delete repeating elements in array.

```
#!/bin/bash

echo "Enter array length:"
read -p "length=>" length
echo "Enetr the array element:"
for((index1=0;index1< $length;index1++))
do
        echo "Enetr array [$index1]:"
        read array[$index1]
done
flag=0
for((index1=0;index1< $length;index1++))
do
        for((index2=`expr $index1 + 1`;index2 < $length;index2++))
        do
                if [ ${array[$index1]} -eq ${array[$index2]} ]
                then
                        flag=1
                        length=`expr $length - 1`
                        for((index3=$index2 ; index3 < $length ; index3++))
                        {
                                array[$index3]=${array[`expr $index3 + 1`]}
                        }
                 else
                        index2= `expr $index2 + 1`
                fi
        done
done

if [ $flag -eq 0 ]
then
        echo "No duplicate"
else
```

```
        for((index1=0;index1<`expr $length - 1`; index1++))
        do
                echo "array[$index1]=${array[$index1]}"
        done
fi
```

8] What is Bash ? What is chmod ? What are all possible modes ? How to chnage the mode ?

9] Write a program to reverse a string using shell scripting.

```
#!/bin/bash
reverse= “ ”
read -p "Enter string:" string
length=${#string}
for (( index=$length-1; index>=0; index-- ))
do
        reverse="$reverse${string:$index:1}"
done
echo "$reverse"
```

10] Write a program to find factorial of given number using shell scripting

```
#!/bin/bash
echo "Enetr number:"
read -p "Number" number
fact=1
var=1
while (( $var <= $number ))
do
        fact=$(( $var * $fact ))
        var= `expr $var + 1`
done

echo "factorial= $fact"
```