**EX NO:2 SHELL PROGRAMMING**

**DATE:04/03/2021**

**Aim:**To Study about the shell programming

**Programs:**

**1.Write a shell program for getting and displaying academic details.Inputs are name,roll no,three marks of students and outputs are name,rollno,total and average.**

**Algorithm:**

**1.** Start

**2.** Create a file using vi command with filename.sh

**3.** Using ‘echo’ command, print the statement “Enter the student details”

**4**. Using ‘read’ command, read name,rollno,mark1,mark2,mark3 from the user

**5.** Using ‘expr’ command, evaluate the total marks by adding mark1,mark2,mark3

and average marks by dividing total marks by total no of marks i.e., 3

**6.** Using ‘echo’ command, print the student’s name,rollno,total marks,average to the user.

**7.** Stop.

**Program:**

echo “Enter the student details”

read name

read rollno

read mark1

read mark2

read mark3

total=`expr $mark1 + $mark2 + $mark3`

average=`expr $total / 3`

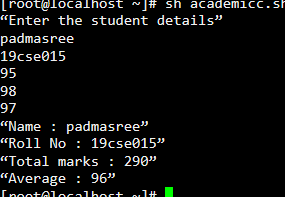
echo “Name : $name”

echo “Roll No : $rollno”

echo “Total marks : $total”

echo “Average : $average”

**Output:**



**2.Write a shell program to implement the arithmetic operations.**

**Algorithm:**

**1.** Start

**2.** Create a file using vi command with filename.sh

**3.** Using ‘echo’ command, print the statement “Enter the value of two numbers:”

**4**. Using ‘read’ command, read the value of a and b from the user

**5.** Using ‘expr’ command, evaluate arithmetic operations such as add, subtract,multiply,divide,modulo the values of a and b

**6.** Using ‘echo’ command, print the addition,subtraction,multiplication,division,moduloto the user.

**7.** Stop.

**Program:**

echo “Enter the value of two numbers”

read a

read b

echo “a = $a”

echo “b = $b”

add=`expr $a + $b`

sub=`expr $a - $b`

mul=`expr $a \\* $b`

div=`expr $a / $b`

mod=`expr $a % $b`

echo “Addition of $a and $b is $add”

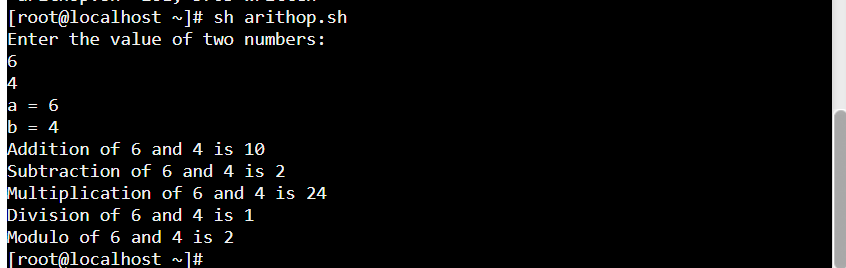
echo “Subtraction of $a and $b is $sub”

echo “Multiplication of $a and $b is $mul”

echo “Division of $a and $b is $div”

echo “Modulo of $a and $b is $mod”

**Output:**



**3.Write a shell program to check whether the given number is positive,negative or zero**

**Algorithm:**

**1.** Start

**2.** Create a file using vi command with filename.sh

**3.** Using ‘echo’ command, print the statement “Enter a number:”

**4**. Using ‘read’ command, read the value of a from the user

**5.**Check whether the number is positive ,negative or zero by using ‘if’ conditonal statements

**6.**After ‘if’ give the condition as $a -gt 0 and next give the keyword ‘then’ and if the condition satisfies ,print the statement as “Given number a is a positive number”

**7.**If the given condition fails,give the condition as $a -lt 0 in ‘elif’ statement and next give the keyword ‘then’ and if the condition satisfies ,print the statement as “Given number a is a negative number”

**8**. If the none of the condition satisfies,print the statement “Given number a is zero” in else statement and give ‘fi’ to terminate the if statements.

**9.**Stop

**Program:**

echo “Enter a number”

read a

if [ $a -gt0 ]

then

echo “$a is a positive number”

elif [ $a -lt0 ]

then

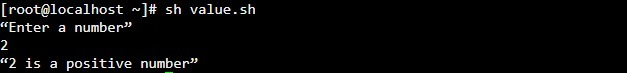
echo “$a is a negative number”

else

echo “$a is zero”

fi

**Output:**

****

**4.Write a shell program to print the combination of 123**

**Algorithm:**

**1**. Start

**2.**Create a file using vi command with filename.sh

**3.**Using 3 ‘for’ loops , we can print the combination of 123

**4.**First give ‘for var in 1 2 3’ and then give ‘do’ keyword ,like wise create 3 for loops inside one another with 3 different variables

**5.** Inside the third loop print the value of three variables

**6.** combination of 123 gets printed

**7.** Using ‘done’ keyword, end the each for loops one after another.

**8.** Stop

**Program:**

for $a in 1 2 3

do

for $b in 1 2 3

do

for $c in 1 2 3

do

echo “$a $b $c”

done

done

done

**Output:**



**5.Write a shell program to find the area of triangle,circle,square and rectangle using case.**

**Algorithm:**

**1.**Start

**2.**Create a file using vi command with filename.sh

**3.**Using ‘echo’ command, print the statement “Enter your choice between 1 and 4:”

**4**. Using ‘read’ command, read the value of a from the user

**5.**Inside case, give 4 cases for triangle,circle,square,rectangle respectively. User can choose any case to execute it.

**6.** In first case, read the values of base and height from user and calculate and print the area of triangle.

**7.** In second case, read the values of radius from user and calculate and print the area of circle.

**8.** In third case, read the values of side from user and calculate and print the area of square.

**9.** In fourth case, read the values of length and breadth from user and calculate and printthe area of rectangle.

**10.** In default case, print “Your choice is out of limit!”

**11.** Terminate case with keyword ‘esac’

**12.** Stop

**Program:**

echo “Enter your choice between 1 and 4:”

read a

case $a in

1)echo “Enter the value of base and height of triangle”

read b

read h

area=`expr $b \\* $h \\* 1 / 2`

echo “Area of a triangle = $area”

;;

2)echo “Enter the value of radius of circle”

read r

area=`expr $r \\* $r \\* 22 / 7`

echo “Area of a circle = $area”

;;

3)echo “Enter the value of side of square”

read s

area=`expr $s \\* $s`

echo “Area of a square = $area”

;;

4)echo “Enter the value of length and breadth of rectangle”

read l

read b

area=`expr $l \\* $b`

echo “Area of a triangle = $area”

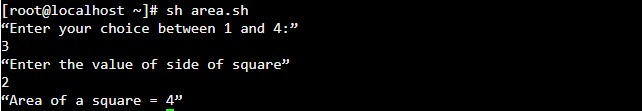
;;

\*)echo “Your choice is out of limit!”

;;

esac

**Output:**



**6.Write a shell program to concatenate two strings and find the length of two strings.**

**Algorithm:**

**1**.Start

**2**.Create a new file using vi command filename.sh

**3**.Get the two strings s1,s2 from the user using ‘read’ command.

**4**.Concatenate two strings by simply joining them and store in in the variable s3.

**5**.Print the concatenated string using ‘echo’ command.

**6.**Print the length of the concatenated string using the statement $s3 | wc -c.

**7**. Using ‘expr’ command calculate length of string and print the length of the concatenated string using ‘echo’ command.

**8**.Stop.

**Program:**

echo "Enter the string1"

read s1

echo "Enter the string2"

read s2

s3=$s1$s2

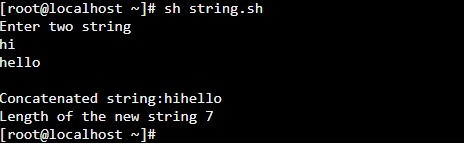
echo "Concatenated string is $s3"

len=`echo $s3 | wc -c`

len=`expr $len - 1`

echo "Length of the concatenated string is $len"

**Output:**



**7.Write a shell program to display the oddposition of the number.**

**Algorithm:**

**1.**Start

**2.**Create a new file using vi command filename.sh

**3.**Get the input numbernum,count from the user using ‘read’ command.

**4**.Declare on more variable n with value as 1.

**5.**Using ‘while’ loop with condition as ‘value of n is less than or equal to value of count’ do the step 6 and 7.

**6.**Print the digit in the ath column digit using ‘echo’ command.

**7.**Increment the value of a by 2.

**8.**Repeat step 6 and 7 until the condition fails.

**9.**Stop.

**Program:**

echo "Enter the number of digits in a number"

read count

echo "Enter a number with $count digits"

read num

n=1

while [ $n -le $count ]

do

a=`echo $num | cut -c $n`

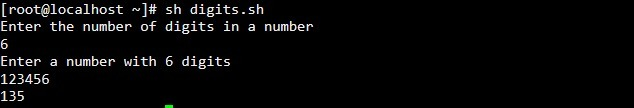
echo -n "$a"

n=`expr $n + 2`

done

echo " "

**Output:**

****

**8.Write a shell program to search an element in array.**

**Algorithm:**

**1.**Start

**2.**Create a new file using vi command filename.sh

**3.**Get the size of array n from the user using ‘read’ command.

**4.**Declare a flag available with value as 1.

**5.**Get the array elements from the user using ‘while’ loop by giving condition ‘value of n not equal to size’ and do the following statements.

**6.**Using ‘read’ command, get the value from user and assign it to the array a[$n].Using ‘expr’ command increment n value by 1. Loops gets terminated when value of n equals to size.

**7.**Get the key element to be searched from the user using ‘read’ command.

**8.**In another ‘while’ loop(which runs upto the size of the array), search for the key in the array using ‘if’ statement.

**9.**In ‘if’ statement, specify the equality condition as ‘if the element at ith position matches the key’, set value of flag as 0.

**10.** Repeat the step 9 until the loop terminates.

**11**.Then using ‘if’ statement, check whether the value of flag matches 1 .

**12.**If it is true, print “Searched element is not found” else print “Searched element is found”.

**13.**Stop.

**Program:**

echo "Enter the size of the array:"

read size

echo "Enter the elements in the array:"

n=0

flag=1

while [ $n -ne $size ]

do

read val

a[$n]=$val

n=` expr $n + 1 `

done

echo "Enter the element to be searched:"

read s

n=0

while [ $n -lt $size ]

do

if [ $s -eq ${a[$n]} ]

then

flag=0

fi

n=` expr $n + 1 `

done

if [ $flag -eq1 ]

then

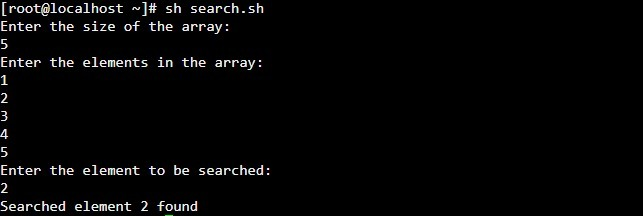
echo "Searched element $s not found"

else

echo "Searched element $s found"

fi

**Output:**



**9.Write a delete zero sized file using if and for.**

**Algorithm:**

**1.**Start

**2.**Create a new file using vi command filename.sh

**3.**Use ‘for’ loop which iterate over all the files in the root directory, and do the following steps.

**4.**Using ‘if’ statement, check whether the file has size using –s command.

**5.**If the file has size greater than zero, print that the “File has size > 0”

**6.**Else print “File is an empty file” and remove(delete) that file using rm command.

**7**.Repeat the steps 4 and 5 until the loop terminates.

**8.**Stop.

**Program:**

for i in \*

do

if [ -s $i ]

then

echo "The file $i has size greater than zero"

else

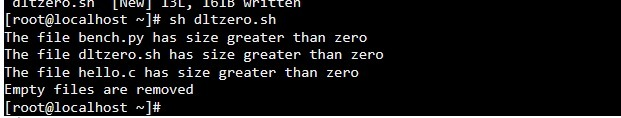
echo "$i is an empty file"

rm -rf $i

fi

done

echo "Empty files are removed"

**Output: **

**10.Write a shell program to reverse a number.**

**Algorithm:**

**1**.Start

**2.**Create a new file using vi command filename.sh

**3.** Get the input number n from the user using ‘read’ command.

**4.**Declare two more variables temp and rev and assign values as 0.

**5.**Using ‘while’ loop with condition ‘while value of n is greater than 0’ do the following steps.

**6.**Using ‘expr’ command, evaluate the modulo of n and 10 and store it in temp.

**7.**Using ‘expr’ command, evaluate the multiplication of rev and 10 and add temp with it and store it back in rev.

**8.**Using ‘expr’ command, evaluate the division of n and 10 and store it in n.

**9.**Repeat the steps 6,7 and 8 until the condition is satisfied.

**10.**Stop.

**Program:**

echo "Enter a number"

read n

rev=0

temp=0

while [ $n -gt0 ]

do

temp=`expr $n % 10`

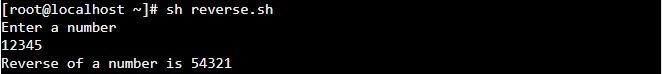
rev=`expr $rev \\* 10 + $temp`

n=`expr $n / 10`

done

echo "Reverse of a number is $rev"

**Output:**



|  |  |
| --- | --- |
| **Observation(20)** |  |
| **Record(5)** |  |
| **Total(25)** |  |
| **Initial** |  |

**Result:**

Thus the above programs were executed and outputs were noted.