PRATHEEK U B
289226

Shell Scripting Project

Palindrome:

```
GNU nano 7.2
echo "Enter the number:"
read n
num=$n
rev=0
while [ $n -gt 0 ]
do
a=`expr $n % 10`
n=`expr $n / 10`
rev=`expr $rev \* 10 + $a`
done
echo $rev
if [ $num -eq $rev ]
then
        echo "The number is a palindrome!"
else
        echo "The number is not a palindrome number!"
fi
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ ./palindrome.sh
Enter the number:
456
654
The number is not a palindrome number!
igris@2b4579b9c7125f2:~/practice_problems$ . palindrome.sh
Enter the number:
1234321
1234321
The number is a palindrome!
igris@2b4579b9c7125f2:~/practice_problems$
```

Pascal:

```
  GNU nano 7.2
pastri()
{
r=$1
c=1
for((i=0;i<r;i++))
{
for((s=1;s<r-i;s++))
{
echo -n "   "
}
for((j=0;j<=i;j++))
{
if [ $j -eq 0 -o $i -eq 0 ]
then
c=1;
else
c=$((c*(i-j+1)/j))
fi
echo -n $c "   "
}
echo
}
}

echo "enter the number of rows:"
read r
pastri $r
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ . pascal.sh
enter the number of rows:
12
                    1
                 1    1
              1    2    1
            1    3    3    1
          1    4    6    4    1
        1    5    10    10    5    1
      1    6    15    20    15    6    1
    1    7    21    35    35    21    7    1
  1    8    28    56    70    56    28    8    1
1    9    36    84    126    126    84    36    9    1
1    10    45    120    210    252    210    120    45    10    1
1    11    55    165    330    462    462    330    165    55    11    1
igris@2b4579b9c7125f2:~/practice_problems$
```

Arithmetic:

```
  GNU nano 7.2                                                    arithmetic
#!/usr/bin/bash
read -p "Input1 : " inp1
if [[ -z $inp1 ]]
then
        echo "Input 1 cannot be empty, plese enter an integer."
        exit
fi

read -p "Input2 : " inp2
if [[ -z $inp2 ]]
then
        echo "Input 2 cannot be empty, please enter an integer."
        exit
fi

bc_val=`echo "$inp1+$inp2" | bc`
echo "BC Value : $bc_val"

expr_val=`expr $inp1 + $inp2`
echo "EXPR Value : $expr_val"
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ . arithmetic
Input1 : 10
Input2 : 20
BC Value : 30
EXPR Value : 30
```

Reverse Nums:

```
  GNU nano 7.2                                          reverseNums.sh
#!/bin/bash

n=12345
rev=0

while [ $n -gt 0 ]; do
    sd=$((n % 10))        # Extract the last digit
    rev=$((rev * 10 + sd)) # Update the reversed number
    n=$((n / 10))          # Remove the last digit
done

echo "Reverse Number is $rev"
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ nano reverseNums.sh
igris@2b4579b9c7125f2:~/practice_problems$ . reverseNums.sh
Reverse Number is 54321
```

Array Sum:

```
  GNU nano 7.2                                          arraysum.sh *
arr=(2 4 -5 -8 9 12)

for (( i = 0; i <= ${#arr[*]}; i++ )); do
  if (( arr[i] > 0 )); then
    sum=`expr $sum + ${arr[i]}`
  fi
done
echo "$sum"
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ nano arraysum.sh
igris@2b4579b9c7125f2:~/practice_problems$ chmod 777 arraysum.sh
igris@2b4579b9c7125f2:~/practice_problems$ . arraysum.sh
27
igris@2b4579b9c7125f2:~/practice_problems$
```

Bubble Sort:

```
  GNU nano 7.2                                          bubbleSort.sh
#bubble sort in shell with static array

declare -a arr
arr=(10 8 20 100 12)

echo "Entered array:"
echo ${arr[@]}

for ((i = 0; i < 5; i++))
  do
    for((j = 0; j < 5-i-1; j++))
      do
        if [ ${arr[j]} -gt ${arr[$((j+1))]} ];
          then
            temp=${arr[j]}
            arr[$j]=${arr[$((j+1))]}
            arr[$((j+1))]=$temp
fi
done
done
echo "Sorted array:"
echo ${arr[@]}
```

OUTPUT:

```
igris@2b4579b9c7125f2:~/practice_problems$ nano bubbleSort.sh
igris@2b4579b9c7125f2:~/practice_problems$ chmod 777 bubbleSort.sh
igris@2b4579b9c7125f2:~/practice_problems$ . bubbleSort.sh
Entered array:
10 8 20 100 12
Sorted array:
8 10 12 20 100
```

Personal Msg:

```
GNU nano 7.2                                              personal_msg.sh
#!/bin/bash

# Prompt the user for their name
echo "Hello! What's your name?"
read name

# Greet the user with a personalized message
echo "Hello, $name! Welcome to the world of shell scripting!"
```

OUTPUT:

```
Hello, IGRIS! Welcome to the world of shell scripting!
igris@2b4579b9c7125f2:~/practice_problems$ nano personal_msg.sh
igris@2b4579b9c7125f2:~/practice_problems$ . personal_msg.sh
Hello! What's your name?
IGRIS
Hello, IGRIS! Welcome to the world of shell scripting!
```