1. To find Largest of Three Numbers

```
read a b c; echo $((a>b && a>c ? a : (b>c ? b : c)))
```

2. To find a year is leap year or not.

```
read y; ((y%4==0 && y%100!=0 || y%400==0)) && echo "Leap" || echo "Not"
```

3. To input angles of a triangle and find out whether it is valid triangle or not

```
#!/bin/bash
echo "Enter three angles of a triangle:"
read a b c
sum=$((a + b + c))
if (( sum == 180 && a > 0 && b > 0 && c > 0 )); then
   echo "Valid triangle"
else
   echo "Invalid triangle"
fi
```

4. To check whether a character is alphabet, digit or special character.

```
#!/bin/bash
echo "Enter a character:"
read ch
if [[ $ch =~ [a-zA-Z] ]]; then
   echo "Alphabet"
elif [[ $ch =~ [0-9] ]]; then
   echo "Digit"
else
   echo "Special Character"
fi
```

5. To calculate profit or loss

```
#!/bin/bash
echo "Enter cost price and selling price:"
read cp sp
if (( sp > cp )); then
   echo "Profit of $((sp - cp))"
elif (( sp < cp )); then
   echo "Loss of $((cp - sp))"
else
   echo "No profit no loss"
```

```
fi
```

6. To print all even and odd number from 1 to 10

```
#!/bin/bash
for i in {1..10}
do
    if (( i % 2 == 0 )); then
        echo "$i is Even"
    else
        echo "$i is Odd"
    fi
done
```

7. To print table of a given number

```
#!/bin/bash
echo "Enter a number:"
read n
for i in {1..10}
do
    echo "$n x $i = $((n * i))"
done
```

8. To find factorial of a given integer

```
#!/bin/bash
echo "Enter a number:"
read num
fact=1
for (( i=1; i<=num; i++ ))
do
    fact=$((fact * i))
done
echo "Factorial of $num is $fact"
```

9. To print sum of all even numbers from 1 to 10.

```
#!/bin/bash
sum=0
for i in {1..10}
do
    if (( i % 2 == 0 )); then
```

```
        sum=$((sum + i))
    fi
done
echo "Sum of even numbers from 1 to 10 is $sum"
```

10. To print sum of digit of any number.

```
#!/bin/bash
echo "Enter a number:"
read n
sum=0
while (( n > 0 ))
do
    digit=$((n % 10))
    sum=$((sum + digit))
    n=$((n / 10))
done
echo "Sum of digits is $sum"
```

11. To make a basic calculator which performs addition, subtraction, Multiplication, Division

```
#!/bin/bash
echo "Enter two numbers:"
read a b
echo "Choose operation (+ - * /):"
read op

case $op in
    +) echo "Result: $((a + b))" ;;
    -) echo "Result: $((a - b))" ;;
    \*) echo "Result: $((a * b))" ;;
    /) echo "Result: $((a / b))" ;;
    *) echo "Invalid operation" ;;
esac
```

12. To print days of a week.

```
#!/bin/bash
days=("Sunday" "Monday" "Tuesday" "Wednesday" "Thursday" "Friday" "Saturday")
```

```
for day in "${days[@]}"
do
    echo $day
done
```

13. To print starting 4 months having 31 days.

```
#!/bin/bash
months=("January" "March" "May" "July")
echo "Months with 31 days:"
for m in "${months[@]}"
do
    echo $m
done
```

14. Using functions.

a. To find given number is Amstrong number or not
b. To find whether a number is palindrome or not
c. To print Fibonacci series upto n terms
d. To find given number is prime or composite
e. To convert a given decimal number to binary equivalent

```
is_armstrong() {
    num=$1
    sum=0
    n=$num
    while (( n > 0 ))
    do
        digit=$((n % 10))
        sum=$((sum + digit ** 3))
        n=$((n / 10))
    done
    if (( sum == num )); then
        echo "$num is an Armstrong number"
    else
        echo "$num is not an Armstrong number"
    fi
}
read -p "Enter a number: " n
```

```
is_armstrong $n
```

(b)

```
is_palindrome() {
   n=$1
   rev=0
   orig=$n
   while (( n > 0 ))
   do
      digit=$((n % 10))
      rev=$((rev * 10 + digit))
      n=$((n / 10))
   done
   [[ $rev -eq $orig ]] && echo "$orig is a palindrome" || echo "$orig is not a palindrome"
}
read -p "Enter a number: " n
is_palindrome $n
```

)(c)

```
fibonacci() {
   n=$1
   a=0
   b=1
   echo -n "$a $b "
   for (( i=2; i<n; i++ ))
   do
      fn=$((a + b))
      echo -n "$fn "
      a=$b
      b=$fn
   done
   echo
}
read -p "Enter number of terms: " n
fibonacci $n
```

(d))

```
is_prime() {
   n=$1
   if (( n < 2 )); then
```

```
        echo "Neither prime nor composite"
        return
    fi
    for (( i=2; i*i<=n; i++ ))
    do
        if (( n % i == 0 )); then
            echo "$n is composite"
            return
        fi
    done
    echo "$n is prime"
}
read -p "Enter a number: " n
is_prime $n
```

(e)

```
decimal_to_binary() {
    n=$1
    bin=""
    while (( n > 0 ))
    do
        bin=$((n % 2))$bin
        n=$((n / 2))
    done
    echo "Binary: $bin"
}
read -p "Enter a decimal number: " n
decimal_to_binary $n
```