1.reverse a no. and check its palindrome or not

//palindrome  
fun main(){  
 val num:Int=12321  
 var rev:Int=0  
 var temp=num  
 while(temp!=0){  
 var d=temp%10  
 rev=rev\*10+d  
 temp=temp/10  
 }  
 if(num==rev){  
 *println*("$rev is equalto $num\n so the number is palindrome ")  
 }  
 else{  
 *println*("$rev is not equalto $num\n so the number is not palindrome")  
 }  
  
}

O/p 12321 is equalto 12321

so the number is palindrome

2.compute the sum of digitsm in given integer

//sum of digits  
fun main(){  
 val num=123  
 var temp=num  
 var sum=0  
 while(temp!=0){  
 var d=temp%10  
 sum=sum+d  
 temp=temp/10  
 }  
 *println*("the sum of digits in the number $num is $sum")  
}

O/P: the sum of digits in the number 123 is 6

3.product of two nos. without using \*

//product of two nos.  
fun main(){  
 var a=5  
 var b=4  
 var pro=a  
 for(i in 1..b-1){  
 pro=pro+a  
 }  
 *println*(pro)  
}

O/P:20

4.check whether the given number is amstrong number

//Amstrong nos or not  
fun pow(b:Int,p:Int):Int{  
 var ans:Int=1  
 for (i in 1..p){  
 ans=ans\*b  
 }  
 return ans  
}  
fun order(no:Int):Int{  
 var temp=no  
 var nd=0  
 while(temp!=0){  
 temp=temp/10  
 nd++  
 }  
 return nd  
}  
  
fun main(){  
 val n:Int=9474  
 var temp=n  
 var ams=0  
 var d:Int=0  
 var nd=*order*(n)  
 while (temp!=0){  
 d=temp%10  
 ams=ams+*pow*(d,nd)  
 temp=temp/10  
 }  
   
 if(ams==n){  
 *println*("the given $n is amstrong no.")  
 }  
 else{  
 *println*("the given $n is not amstrong no.")  
 }  
}

O/P:

the given 9474 is amstrong no.

5.Check whether the a given no. is perfect number

fun main(){  
 val n= *readln*().*toInt*()  
 var sum=0  
 for(i in 1..n-1){  
 if(n%i==0){  
 sum=sum+i  
 }  
 }  
 if (n==sum){  
 *println*("the given number $n is a perfect number")  
 }  
 else{  
 *println*("the given number $n is not a perfect number")  
 }  
}

O/P:

28

the given number 28 is a perfect number

6.print amstrong no. between 1to1000

fun pow(b:Int,p:Int):Int{  
 var ans:Int=1  
 for (i in 1..p){  
 ans=ans\*b  
 }  
 return ans  
 }  
fun order(no:Int):Int{  
 var temp=no  
 var nd=0  
 if (temp-9<=0){  
 nd=0  
 }  
 else {  
 while (temp != 0) {  
 temp = temp / 10  
 nd++  
 }  
 }  
 return nd  
}  
  
fun main(){  
 var high=1000  
 var low=1  
 for (i in low..high) {  
 var n = i  
 var temp = n  
 var ams = 0  
 var d: Int = 0  
 var nd = *order*(n)  
 while (temp != 0) {  
 d = temp % 10  
 ams = ams + *pow*(d, nd)  
 temp = temp / 10  
 }  
 if (n == ams) {  
 *println*(n)  
 }  
 }  
}

O/P:

1

153

370

371

407

7.generate fibonacci series using command line argument

8.compute fibonacci series using command line argument

fun main(){  
  
 var a=0  
 var b=1  
 var sum=0  
 val n= *readln*().*toInt*()  
 *print*("$a $b")  
 for (i in 2..n){  
 sum=a+b  
 a=b  
 b=sum  
 *print*(" $sum")  
 }  
  
}

O/p:

8

0 1 1 2 3 5 8 13 21

9.calculate sum and average of an array

//sum and avg of array  
  
fun main(){  
 var a= *arrayOf*(1,2,3,4,5,6,7,8,9)  
 val n=a.size  
 var sum:Int=0  
 for (i in 0..n-1){  
 sum=sum+a[i]  
 }  
 var avg=sum/n  
  
 *println*("Sum is $sum \n avg is $avg")  
  
}

O/p:

Sum is 45

avg is 5

10.Find the second largest and smallest element in array

//2nd largest and smallest  
  
fun main(){  
 var a= *arrayOf*(3,4,34,24,78,99,0,75,12,23,41)  
 val n=a.size  
 var temp=0  
 for (k in 0..n-1) {  
 for (i in 0..n - 2) {  
 var j = i + 1  
 if (a[i] > a[j]) {  
 temp = a[i]  
 a[i] = a[j]  
 a[j] = temp  
 }  
 }  
 }  
 *println*("Second largest element is ${a[n-2]}\nSecond smallest element is ${a[1]} ")  
}

O/P:

Second largest element is 78

Second smallest element is 3

11.calculate sum of even and odd nos. in array

//Sum of odd and even  
  
fun main(){  
  
 var a= *arrayOf*(12,45,76,89,3,43,44,23,12)  
 var odd:Int=0  
 var even:Int=0  
 val n=a.size  
 for (i in 0..n-1){  
 if(a[i]%2==0){  
 even=even+a[i]  
 }  
 else{  
 odd=odd+a[i]  
 }  
 }  
 *println*("Sum of even is $even\n Sum of odd $odd")  
}

O/P:

Sum of even is 144

Sum of odd 20

12.check two elements in the array such that difference between them is largest

//two elements in the array such that difference between them is largest  
  
fun main() {  
  
 var a = *arrayOf*(12, 45, 76, 89, 3, 43, 44, 23, 12)  
 var b= *arrayOf*(2,4,53,1,6,87,54,32,12,45,78)  
 val an=a.size  
 var bn=a.size  
  
  
 var temp=0  
 for (k in 0..an-1) {  
 for (i in 0..an - 2) {  
 var j = i + 1  
 if (a[i] > a[j]) {  
 temp = a[i]  
 a[i] = a[j]  
 a[j] = temp  
 }  
 }  
 }  
  
 temp=0  
 for (k in 0..bn-1) {  
 for (i in 0..bn - 2) {  
 var j = i + 1  
 if (b[i] > b[j]) {  
 temp = b[i]  
 b[i] = b[j]  
 b[j] = temp  
 }  
 }  
 }  
 var dif1=(a[an-1]-b[0])  
 var dif2=(b[bn-1]-a[0])  
 if(dif1>dif2){  
 *println*(dif1)  
 }  
 else{  
 *println*(dif2)  
 }  
}

O/P:

88

13.Interchange any two rows and column in the matrix

//.Interchange any two rows and column in the matrix  
  
fun main() {  
 var mat = *arrayOf*(*intArrayOf*(1, 2, 3), *intArrayOf*(4, 5, 6), *intArrayOf*(7, 8, 9))  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[i][j]} ")  
 }  
 *println*()  
 }  
 *println*("ENTER YOUR CHOICE \n1.INTERCHANGE ROWS\n2.INTERCHANGE COLUMNS")  
 var ch: Int = Integer.valueOf(*readLine*())  
  
 if (ch == 1) {  
 *println*("enter the row want to change: ")  
 var r = Integer.valueOf(*readLine*())  
 *println*("enter the row want to change with: ")  
 var r1 = Integer.valueOf(*readLine*())  
 for (j in 0..mat.size - 1) {  
 var temp = mat[r][j]  
 mat[r][j] = mat[r1][j]  
 mat[r1][j] = temp  
 }  
 *println*("After interchange")  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[i][j]} ")  
 }  
 *println*()  
 }  
 }  
 else if(ch==2) {  
 *println*("enter the column want to change: ")  
 var c = Integer.valueOf(*readLine*())  
 *println*("enter the column want to change with: ")  
 var c1 = Integer.valueOf(*readLine*())  
 for (j in 0..mat.size - 1) {  
 var temp = mat[j][c]  
 mat[j][c] = mat[j][c1]  
 mat[j][c1] = temp  
 }  
 *println*("After interchange")  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[i][j]} ")  
 }  
 *println*()  
 }  
  
 }  
 }

O/P:

1 2 3

4 5 6

7 8 9

ENTER YOUR CHOICE

1.INTERCHANGE ROWS

2.INTERCHANGE COLUMNS

2

enter the column want to change:

0

enter the column want to change with:

2

After interchange

3 2 1

6 5 4

9 8 7

14.Find the tranpose of a matrix

//TRANPOSE OF THE MATRIX  
  
fun main() {  
 var mat = *arrayOf*(*intArrayOf*(1, 2, 3), *intArrayOf*(4, 5, 6), *intArrayOf*(7, 8, 9))  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[i][j]} ")  
 }  
 *println*()  
 }  
 *println*("Transposed matrix")  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[j][i]} ")  
 }  
 *println*()  
 }  
}

O/P:

1 2 3

4 5 6

7 8 9

Transposed matrix

1 4 7

2 5 8

3 6 9

15.Display Upper triangular matrix

fun main(){  
  
 var a= *arrayOf*(*arrayOf*(1,2,3,4,5), *arrayOf*(0,5,6,5,7), *arrayOf*(0,0,9,1,9), *arrayOf*(0,0,0,5,2), *arrayOf*(0,0,0,0,5))  
 val row=a.size-1  
 val col=a[0].size-1  
 var flag=0  
 for (i in 0.. row){  
 for (j in 0..col){  
 *print*("${a[i][j]} ")  
 }  
 *println*()  
 }  
  
  
 for (i in 1.. row){  
 for (j in 0..col-1) {  
 if (i==j){  
 continue  
 }  
 if(i<=row/2 && j>=col/2){  
 continue  
 }  
 if (a[i][j]!=0){  
 flag++  
 }  
  
 }  
  
 }  
 if (flag==0){  
 *println*("the given matrix is upper triangular matrix")  
 }  
 else {  
 *println*("the given matrix is not upper triangular matrix")  
 }  
}

O/P:

1 2 3 4 5

0 5 6 5 7

0 0 9 1 9

0 0 0 5 2

0 0 0 0 5

the given matrix is upper triangular matrix

16.Frequency of odd and even nos. in given matrix

//Frequency of odd and even nos. in given matrix  
  
  
fun main() {  
 var mat = *arrayOf*(*intArrayOf*(1, 2, 3), *intArrayOf*(4, 5, 6), *intArrayOf*(7, 8, 9))  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 *print*("${mat[i][j]} ")  
 }  
 *println*()  
 }  
 var even:Int=0  
 var odd:Int=0  
 for (i in 0..mat.size - 1) {  
 for (j in 0..mat[0].size - 1) {  
 if(mat[i][j]%2==0)  
 {  
 even++  
 }  
 else  
 {  
 odd++  
 }  
 }  
 }  
 *println*("number of odd nos. in the matrix is $odd")  
 *println*("number of even nos. in the matrix is $even")  
}

O/P:

1 2 3

4 5 6

7 8 9

number of odd nos. in the matrix is 5

number of even nos. in the matrix is 4