06/06/23

Arrays

→ Sum of even array elements and product of odd indexed elements.

```
Package model3method;
Public class arraysum{
  int sum=0, prod=1;
  Public int arr(int a[]){
   for int arr(int i=0; i<a.length;i++){
     if (a[i]%2==0){
          Sum=sum+a[i];
     }
   } return sum; }
Public int oddindex (int a[]){
   for (int i=0; i<a.length; i++){
       if (i%2!=0){
           Prod=prod*a[i];
       }
     return prod; }
Public static void main (string args[]){
    int a[]=new int[] {1,2,3,4};
    arraysum b=new arraysum();
    syso ("sum of numbers":+b.arr(a));
    Syso ("product of a numbers:"+b.oddindex(a));
  }
}
```

→ int a= [1,2,3,4] reverse of a should be copied in another array as b[]={4,3,2,1};

```
Package arrays;
Public reverse{
    Public static void main(String args[]){
    int a[]={1,4,2,6}, temp=0;
    int b[] = new int [a.length];
for(int x=0 ; x<a.length ; x++){ syso(a[x]); }
for (int y= a.length-1; y>0; y--){
    b[temp]=a[y];
    temp++; }
for (int z=0; z<b.length; z++){
    syso(b(z));
```

In case square of reverse [36 4 8 1]

```
    b[temp]=a[y].a[y];
```

→ Linear Search ★★★★★

```
Package arrays
Public class Searching{
    Psvm(string[] args){
    int a[]={1,2,3,10,9,8,7,6,5,4};
    int flag=10;
for(int x=0; x<a.length; x++){
    if(a[x]==flag){ b=true; }
if(b==true){
    Syso("element found");
else
    syso("Not found");
```

07/06/23

Scanner Array

```
Public class scannersarray{
    Psvm(String args[]){
    Scanner sc= new Scanner(System.in);
     int size =5;
     Syso("please enter the values");
     int a[]= new int[5];
for(int i=0; i<a.length;i++){
        a[i]=sc.nextInt();}
for(int i=0; i<a.length;i++){
        syso(a[i]);}}}
```

→ Midfirstreverse / second reverse - package arrays.

```
Public class midfirstreverse{
    Psvm(String args[]){
  Public void midsecondreverse(int a[]){
     Syso("second reverse");
  for(int i=0;i<=(a.length-1)/2;i++){
        Syso(a[i]);}
for(int i=a.length-1;i>=(a.length-1)/2)+1;i--){
        Syso(a[i]);}
}
Public void midfirst(int a[]){
     Syso("first reverse");
for(int i=(a.length-1)/2;i>=0;i--){
        Syso(a[i]);}
```

```
for (int i=((a.length-1)/2)+1; i<a.length;i++){
    Syso(a[i]); }}
Pvsm (string [] args){
    int a[] = {10,20,30,40,50,60,70};
  midfirstreverse m = new midfirstreverse();
   m.midfirst(a);
   m.midsecondreverse(a); }}
```

→ Random Arrays

```
Public class Randomoo{
Public void method1 (int a[]){
  for(int i=0; i<=(a.length-1)/2;i++){
      Syso (a[i]+" "); }}
Public void method2 (int a[]){
  for(int i=a.length-1; i>(a.length-1)/2;i--){
     Syso (a[i]+" "); }
Pvsm (string args[]){
   Random r= new Random();
    int a[] = new int[6];
for(int i=0;i<a.length; i++){
      a[i]= r.nextInt(100); }
for (int i=0; i<a.length; i++){
     syso(a[i]); }
  syso();
Syso ("Random Reverse"+" ");
randomoo r1 = new randomoo();
   r1.method1(a);
   r1.method2(a);
```

→ Copy array Package arrays.

```
Public class copyarray{
 Public void copy (int a[], int b[]){
      syso("copying");
 for(int i=0; i<a.length;i++){
      b[i]=a[i]; }
 for(int i=0; i<b.length;i++){
      syso(b[i]); }
 Public void problem2(int a[], int b[]){
    Syso(" First Reverse"
 for(int i=(a.length-1)/2; i>0;i--){
      b[i]=a[i]; }
 for(int i=(a.length-1)/2; i>0;i--){
      Syso(b[i]);
 for(int i=((a.length-1)/2)+1; i<a.length;i++){
      b[i]=a[i]; }
 for(int i=((a.length-1)/2)+1;i<a.length;i++){
      Syso(b[i]);}
Public void Problem3 {int a[], int b[]){
    syso("second Reverse");
  for(int i=0; i<=(a.length-1)/2;i++){
      b[i]=a[i];
  }
 for(int i=0;i<=(a.length-1)/2; i++){
      Syso(b[i]); }
```

```
for(int i=a.length-1 ; i>=((a.length-1)/2)+1; i--){
        b[i]= a[i]; }
for(int i = a.length-1; i>=((a.length-1)/2)+1; i--){
        Syso(b[i]); }}
Psvm (String args[]){
    int a[]= {10,20,30,40,50,60,70};
    int b[]= new int[a.length];
    copyarray c=new copyarray();
    c.copy (a,b);
    c.problem2 (a,b); c.problem3(a,b); }}
```

→ Even odd index :- (Arrays)

```
Public class evenodd{
Psvm(String args[]){
 int a[]={1,2,3,4,5,6};
 int b[] =new int [a.length];
for(int i=0; i<a.length; i++){
      b[i]=a[i];
  }
for(int i=0;i<b.length;i++){
    if(i%2==0){
        syso(b[i]); }}
for(int i=0; i<b.length; i++){
      if(i%2!=0){
        syso(b[i]); }
  }
}
```

```
int a={1,2,3,4,5,6}
      0 1 2 3 4 5
even = 1,3,5
odd = 2 4 6
output 1,3,5,2,4,6.
```

Previous next

a[] = {1,2,3,4,5}

→ 1 has no previous value so it should add with next value Eg:- 1 → 1+2=3.

→ 5 has no next value so it should add with previous value Eg:- 5 → 4+5=9.

→ 2 has previous, next so a[i+1]+a[i-1].

```
Package arrays;
Public class previousnext{
    Psvm(String args[]){
        int a[] = {1,2,3,4,5};
        int b[] = new int [a.length];
    for (int i=0; i<a.length ;i++){
            b[i]=a[i];}
        int x=0;
    for(x=0;x<b.length;x++){
        if(x==0){
            b[x]= a[x]+a[x+1];
        }
    else if (x==a.length-1){
            b[x] = a[x] +a[x-1];}
    else{
            b[x] = a[x-1]+a[x+1];}}
for( x=0; x<a.length; x++){
        Syso(b[x]);}}}
```

→ a={1,2,3,4,5,6,-1,0}, flag=5.
{1,4}, {2,3}, {5,0}, {6,-1}

Sol:-
```
Package arrays;
Public class addsamenumber{
    Psvm(String args[]){
    int a[] = {1,2,3,4,5,6,-1,0};
    int flag=5
for(int i=0; i<a.length; i++){
for(int j=i+1; j<a.length; j++){
    if(a[i]+a[j]==flag){
    Syso(a[i]+" "+a[j]); } } } }
```

08/06/23 Prime Or Not Using Arrays and copying in another array.

```
Package arrays;
Public class prime{
    Psvm(String args[]){
    int a[] = new int[]{1,2,3,4,5,6,7,8,9};
    int count=0, count1=0, x=0;
for(int i=0; i<a.length; i++){
    count=0
    for(int j=1; j<a[i]; j++){
        if(a[i]%j==0)
        Count++;
    }
    if(count==2){
        Count1++;
        b=new int[coun+1];
        b[x]=a[i];
        Syso(b[x]);
        x++;
} } }
```

2) int a = {1,2,3,4}, b = {5,6,7,8}.

int c = {1,2,3,4,5,6,7,8} → output.

```
Package arrays;
Public class combine2{
    int i,j;
Public void com (int a[], int b[], int c[]){
    for (i=0; i<a.length; i++){
        c[i]=a[i];
    for (i=0; i<a.length; i++){
        Syso(a[i]);
    }
for (j=0; j<b.length; j++){
        c[i]=b[i]; }
for (j=0; j<b.length; i++){
        Syso(c[j]); }}
PSVM (String[] args){
    int a[]={1,2,3,4};
    int b[]={5,6,7,8};
    int c[]=new int[a.length*2];
    Combine2 d=new combine2();
    d.com(a,b,c); }}
```

→ package arrays;
import java.util.Arrays;
Public class combinezalternate{
pvsm(String args[]){
int a[]={1,2,3,4};
int b[]={5,6,7,8};
int d[]=new int [a.length+b.length];
int x,y=0;
for(x=0;x<a.length;x++){
d[y++]=a[x];
d[y++]=b[x]; } Syso(Arrays.toString(d));}
}

d[]={1,5,2,6,3,7,4,8};

3) Max, Min.

Package arrays;
Public class maxmin{
int min=9;max=0;
Public void maximum(int a[]){
for(int i=0;i<a.length;i++){
if(a[i]>max){
max=a[i];}}
syso(max);}
Public void minimum(int a[]){
for(int i=0;i<a.length;i++){
if(a[i]<min){
min=a[i];}}
syso(min);}

```
PSVM(String args[]){
    int a[]={10,20,30,1,9};
    maximin m= new maximin();
    Syso("maximum");
    m.maximum(a);
    Syso("minimum");
    m.minimum(a);}}
```

Arrays.Sort, Arrays.equal

```
Package arrays;
Public class equals{
    PSVM(String args[]){
      int a[]={10,20,30,40,50};
      int b[]={10,20,30,40};
  for(int i=0;i<a.length;i++){ }
    if(Arrays.equals(a,b)){
          Syso("equals");}
    else{ Syso("Not equals");}}}
```

Arrays:-

```
for(int i=0;i<a.length;i++){ }
Array.sort(a);
```

Palindrome.

```
Public class palindrome{
  PSVM(String args[]){
    int a[] = {1,2,32,121,9};
    int rem=0, rev=0, i, j, y;
  for (i=0; i<a.length; i++){
      rev=0;
    for (y=a[i]; y>0; y/=10){
      rem= y%10;
      rev= rev*10+rem;
    }
    if (rev==a[i]){
      Syso(a[i]); } } }
```

Armstrong.

```
Public class armstrong{
  PSVM(String args[]){
    int a[]={10,20,32,153,1,9};
    int count=0, rem=0, p=0, max=0;
  for(int i=0; i<a.length; i++){
    int y=a[i];
    count=0; p=0;
  for (int j=a[i]; j>0; j=j/10){
      count++; }
  for (int j=a[i]; j>0; j=j/10){
      rem= j%10;
      p=(int) (Math.pow(rem, count))+p; }
    if (p==y){
      Syso(y); } } }
```

Rotations

Left, Right rotations:-

Package rotation:-

```
Public class leftrotation{
  Public void left (int a[], int r){
    for(int i=r;i<a.length; i++){
        syso(a[i]+" ");
    }
    for (int i=0 ;i<r; i++){
        syso (a[i] +" ");
    }
}
Public void right (int a[], int r){
  for(int i= a.length-r; i<a.length; i++){
      syso(a[i])+" "); }
  for (int i=0; i<a.length-r; i++){
      syso (a[i]+" "); } }
Psvm (string args[]){
    int a[]= {10,20,30,40,50,60,70,80,90,100};
    int r=1;
  leftrotation l=new leftrotation();
    Syso("Left rotation");
    l.left(a,r);
    Syso("Right rotation");
    l.right(a,r);

}}
```

→ Storing the left rotate using only one value at backside.

Eg:- a={10,20,30,40,50};

a=1

b={20,30,40,50,10};

```
int temp=a[0];
for(x=0;x<a.length-1;x++){
    a[x]=a[x+1];
}
a[x]=temp;
for(int var:a){
    Syso(var);}
```

Storing the left rotate

```
Public class storing rotations{
  Psvm(string args[]){
int a[]={10,20,30,40,50,60,70,80,90,100};
int r=2,x,temp;
for(x=0;x<r;x++){
for(x=0;x<a.length-1;x++){
    a[x]=a[x+1];}
  a[a.length-1]=temp;}
for(int var:a){
    Syso(var);}
```

Storing the right rotate.

```
int a[] = {1,2,3,4,5}; r=1, i, j;
for (i=0; i<r; i++){
    int temp = a[a.length-1];
for (j = a.length-1; j>=0; j--){
    a[j] = a[j-1];
}
a[0] = temp;
for (int var : a){
    syso(var); }
```

Duplicate Values rotation

```
Public class duplicate values {
  Psvm (string args[]){
    int a[] = {1,1,2,2,3,4,4,5,6,5,7};
    for(int i = 0; i<a.length; i++){
    for(int j=i+1; j<a.length; j++){
        if (a[i] == a[j]){
            Syso(a[i]); }
    }
  }
}
```

Storing the values of Positive (or) Negative

```
Public class positiveNegative {
    PsvM (String[] args) {
    int a[] = {-1,5,-8,2,7,-3,6};
    int b[] = new int[a.length];
    int x=0, y=0;
for (int i=0; i<b.length; i++) {
        if (a[i]<0) {
            b[i] = a[i];
            x++;
        }}
    y = x;
for (int j=0; j<b.length; j++) {
        if (a[j]>0) {
            b[y] = a[j];
            y++;
        }}
for (int k=0; k<b.length; k++) {
        Syso(b[k]); }}
```

Reverse Array

```
Public class reversearray {
    PSVM (String args[]) {
    int a[] = {1,2,3,4,5}, count=4, i, temp;
for (int i=0; i<a.length; i++) {
        temp = a[i];
        a[i] = a[count];
        a[count] = temp;
        count--;
    }
for (int var:a) {
        Syso(var); }}}
```

Deletion (Left rotation)

```
int a[] = {1,2,3,4,5,6,7,8,9,10};
for (int i=9; i<a.length-1; i++){
        a[i] = a[i+1];
    }
for (int var : a){
        Syso(var);
    }
```

Package arrays.

Insertion.

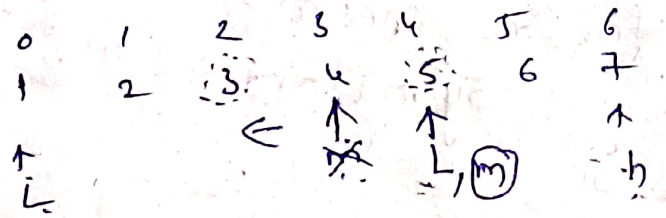
```
int a[] = {10,20,30,40,50,60,70,80,90,100};
    int element = 55, i;
for (i=a.length-1; i>5; i--){
        a[i]=a[i-1];
    }
    a[i] = element;
for(int var : a){
        Syso(var);
    }
```

using Scanner size is different and assigning all the values.

```
Package arrays;
Public class sizedefiningInsertion {
    Public Static void main(String args[]){
      Scanner sc = new Scanner(System.in);
         int a[] = new int[11];
      int i, k, element=55;
    for (int i=0; i<a.length; i++){
           a[i]=sc.nextInt();
      }
    for (int j=0; j<a.length; j++){
          Syso(a[j]+" "); }
      Syso();
    for (k= a.length-1; k>5; k--){
          a[k] = a[k-1];
      }
     a[k] = element;
    for (int var: a) {
          Syso(var+" ");
     }
}
}
```

Binary Search

```
int a[] = {1,2,3,4,5,6,7,8};
 int low=0, high=a.length-1, mid, search=2;
while (low<=high){ mid=(low+high)/2;
    if (a[mid] <search) {
        low = mid+1;
    }
else if (a[mid] > search){
        high=mid-1;
    }
else {
    Syso("element found");
        break;
    }
}
if (low>high){
    Syso("Not element found");
}
}
```



search = 5

m = L+h/2 = 0+6/2 = 3

a[m] < search

4 < 5

L = m+1

m = 6+4 = 5

a[m] = = search

Traditional sorting

```
int a[] = {9, 8, 5, 4};
for (int i=0; i<a.length-1; i++){
    if(a[i] > a[i+1]){
        temp = a[i];
        a[i] = a[i+1];
        a[i+1] = temp;
        i=-1;
    }
}
for (int var: a){ syso(var); }
```

Selection Sort

```
Public class SelectionSort {
    psvm(String args[]){
    int a[] = {1, 2, 4, 2}; i, x, y, temp=0;
                0  1  2  3
    for(x=0; x<a.length; x++){
        i=x;
    for (y=x+1; y<a.length; y++){
        if (a[i] > a[y]){
            i=y;
        }
    }
    temp = a[i];
    a[i] = a[x];
    a[x] = temp;
}
for(int var: a){
    Syso(var);
}
} }
```

Bubble sort

```
Public class bubblesort{
    Public static void main(string[] args){
    int a[]={12,17,19,13,14,23,53,24};
```

Duplicate elements

```
int a[]={10,20,30,30,40,50,60,70,80,70,30,20};
  int i,j,count;
boolean b[]=new boolean[a.length];
for(i=0;i<a.length;i++){
      count=1;
    if(b[i]==true;
      Continue;
  for(j=i+1;j<a.length;j++){
      if(a[i]==a[j]){
        count++;
        b[j]=true;
      }}
  if(count>1){
        syso(a[i]+"->"+count);}}}
```

Unique elements

```
int a[]={10,20,30,30,40,50,60,70,80,70,30,20};
   int i,j,count;
boolean b[]=new boolean[a.length];
for(i=0; i<a.length;i++){
      Count=1;
   if(b[i]==true){
      Continue;
for(j=i+1;j<a.length;j++){
      if(a[i]==a[j]){
        count++;
        b[j]=true;
   }}
if(count==1){
     syso(a[i]+"->"+count);}}}
```

16/06/23 Intersection of two arrays.

```
Public class Mahe{
    psvm(String args[]){
for(int i=0;i<a.length;i++){
    int i=0;
    int j=0;
while(i<arr.length && j<arr.length){
    if(arr1[i]==arr2[i]{
        syso(arr[i]);
        i++; j++;
    }
    else if(arr1[i]>arr2[j]){
        j++; }
    else{
        i++; } } }
```

Second Largest Array :-

```
int a[]={10,20,23,14,17,19,26};
    f=a[0]; s=a[0];
for(x=1; x<a.length; x++){
    if(f<a[x]){
        s=f;
        f=a[x];
    }
elseif(a[x]<f && a[x]>s){
        s=a[x];
    }
}
else if(s==f){
    syso(s);
}
syso(s);
```

2D Array Scanner

```
Public class Make{
 Public static void main(String args[]){
  Scanner sc= new scanner(System.in);
   int row=2; col=3;
  int a[] = new int [2][3];
  for(int i=0; i<row; i++){
  for(int j=0; j<col ; j++){
      a[i][j]= sc.nextInt();
  }
 }
for (int i=0; i<a.length; i++){
 for(int j=0; j<a.length; j++){
    syso(a[i][j]);
  }
```

Sum

```
Public class Make{
  Public static void main(String args[]){
   Scanner sc = new scanner(system.in);
  int row=2, col=3;
  int a[] = new int[row][col];
  int b[] = new int[row][col];
  int c[]= new int[row][col];
```

```
for (int i=0; i<a.length; i++) {
for (int j=0; j<a.length; j++) {
    c[i][j] = a[i][j] + b[i][j];
  }
}
for (int i=0; i<a.length; i++) {
for (int j=0; j<a.length; j++) {
    syso(c[i][j]);
  }
}
```

~~Diagonal~~ Transpose

```
int a[][] = {{1,2,3},{4,5,6},{7,8,9}};

for (int i=0; i<a.length; i++) {
for (int j=0; j<a.length; j++) {
    syso(a[j][i]);
  }
}
```

Left diagonal

```
int a[][] = {{1,2,3},{4,5,6},{7,8,9}};
for (int i=0; i<a.length; i++) {
for (int j=0; j<a.length; j++) {
        if(i == j) {
    syso(a[i]);
```

Right diagonal

```
row=2, col=3;
int a[][] = new int [row][col];
for (int i=0; i<row; i++){
for (int j=0; j<col; j++){
    a[i][j] = sc.nextInt();
  }
}
for (int i=0; i<row; i++){
for (int j=0; j<row; j++){
    if (i+j=col-1){
        syso(a[i]);
    }
  }
  syso()
}
```

→ Delete Duplicate elements in the array.

Second largest

```
Public class Secondlargest {
  Public static void main (string args[]){
    int a[]={80,60,7,8,90};
    int i,j,f=a[0],s=a[0];
  for(int i=1;i<a.length;i++){
      if(f<a[i];
        s=f;
        f=a[i];
      }
    else if(s<a[i]. && f>a[i]){
        s=a[i];
    }
  if(s==f){
      s=a[i]; }}
  Syso(s); }}
```

Strings

lower to upper & upper to lo

→ public class Convertuppertolowwer {

String s = new String("Hello all Good Moming");

for (int i=0; i<s.length(); i++){

if (s.charAt(i) >='A' && s.charAt(i) <='z')

{

Syso((char)(s.charAt(i) + 32) + " ");

}

if (s charAt(i) >='a' && s charAt(i) <='z')

{

Syso (cchar)(s.charAt(i)-32) + " ");

}

}

}

Hi Hello

PALINDROME

Public class palindrome {

Public static void main(String args()) {

String s= new string("MADAM");

string s1 = " ";

int i;

for (i=s.length()-1; i >=0; i--) {

s1 = s1 + s.charAt(i);

}

if(s1.equals(s)){

Syso ("palindrome");

}

else {

Syso ("Not palindrome"); }}}

→ Reverse a String. String problems

```
Public class reverse{
 Psvm (string args[]){
  String s=new string("Hello maha how are you");
  String s1="";
for(int i=s.length()-1; i>0; i--){
        s1=s1+s.charAt(i);
  }
 Syso(s1);}}
```

Spaces count

```
Public class space{
    Psvm (string args[]){
  int count=0;x,y=0
  String s1=new String("--hello- all--Maha--");
  String s1= s.trim();
for(x=0; x<s1.length; x++){
     if(s.charAt(x)==' '){
      if(s.charAt(x-1)!=' ')
        count++
      }
   }
  Syso(count+1);
 }
}
```

Vowels consonants

```
Public class vowelsconsonants {
  Psvm (string args[]){
  String s=new String("hello all good morning");
  int vowels=0, consonants=0;
for(int i=0;i<s.length(); i++){
if(s.charAt(i)=='a' || s.charAt(i)=='e' || s.charAt(i)
=='o' || s.charAt(i)=='u'){
      vowels++;
      syso(s.cha
```

Vowels first consonants next

```
Public class vowelsinone{
  psvm(string args[]){
    String s=("hello");
    string s1="";
    String s2="";
for(int i=0; i<s.length();i++){
if(s.charAt(i)=='a' || s.charAt(i)=='e' || s.charAt(i)=
='o' || s.charAt(i)=='u'){
      s1=s1+s.charAt(i);}}
for(int j=0;j<s.length();j++){
 if(s.charAt(j)!='a' && s.charAt(j)=='e' && s.charAt(j)=
  && s.charAt(j)=='o' && s.charAt(j)=='u'){
      s2=s2+s.charAt(j);
  }
 }
 syso(s1+s2);
}
}
```

→ sort the character array.

```
Package stringproblems;
Public class sortingstrings {
 PSVM (String[] args){
String s = new String ("hello all good morning");
char[] a = s.toCharArray();
 Char temp = a[0];
for(int i=0; i<a.length-1; i++) {
      if (a[i] > a[i+1]){
      temp=a[i];
      a[i] =a[i+1];
      a[i+1] = temp;
       i=-1; } }
 for (char var : a) {
      syso(var); } } }
```

(or)

```
Public class Ascendingorder_String {
 PSVM (String args[]){
 String s1= "Hello all good morning ";
 String s2 = " "; //"";
for(int x=0; x<s2.length; x++){
for(int y=0; y<s2.length-x-1; y++){
 if(s2[y].compareToIgnoreCase(s2[y+1]) >0) {
        String temp = s2[y];
        s2[y] = s2[y+1];
        s2[y+1]= temp; } } }
 for(String var : s2){
        Syso(var);
 } } }
```

Hello
Hello
Hello
Hi
Hello
Hi!

String Buffer

```
StringBuffer sb = new StringBuffer("Helloall");
sb.append("how are you");
```

Index highest no of length

```
Public class highestlength {
  Psvm (string args[]) {
String s = new string ("hello maheswari how
are you");
 String [] a = s.spilt(" ");
for (int i=1; i<a.length; i++) {
    if (max < a[i].length()) {
        index=i
        max = a[i].length(); } }
  Syso (a[index]+" "+max); } }
```

Odd index Reverse.

```
Public class oddindexReverse {
   psvm (string args[]) {
 string s1=new string ("hello all good morning
whatsapp");
string s2 = s1.spilt(" ");
for (int i=0; i<s2.length;i++){
   if (i%2!=0) {
      string s4=s2[i];
for(int j = s4.length()-1; j>=0;j--){
```

```
        Syso(s4.charAt(j));
    }
    Syso(" ");
}
else {Syso(s2[i]);} Syso(" ");
}}}
```

Buffering using Reverse.

```
Public class bufferings{
    PSVM(String[] args){
StringBuffer sb = new StringBuffer("hello");
 syso(sb.append("make"));
String s = new String("Maheswari");
String s1 = new StringBuffer(s).reverse().toString();
 syso(s1)
```

Anagram

```
Public class Anagram{
    PSVM(String args[]){
    String s1 = new String("Listen");
    String s2 = new String("Silent");
char c1[] = s1.touppercase().tocharArray();
char c2[] = s2.touppercase().tocharArray(){
 if(s1.length() == s2.length(){                x));
        Arrays.sort(c1);
       Arrays.sort(c2);
      if(Arrays.equals(c1,c2)){
            Syso("Anagram");
      }
    else{
        Syso("Not Anagram");}
else{
    Syso("Not Anagram");}}
```

20/06/23 Strings.

→ Characters to string.

```
Public class stringtoint {
  Char c[] = {'a','b','c','d'};
  String s = " ";
for (int i=0; i<c.length; i++){
      s = s+a[i];
  }
}
```

String

→ Character to int sum

```
Public class stringtoint {
  psvm (string args[]){
String s = new string ("a1b2c9"); int sum=0
for (int i=0; i<s.length(); i++) {
if (s.charAt(i)>='0' && s.charAt(i)<='9'){
      sum = sum + character.getNumeric
Value (s.charAt(x));
  }
}Syso (sum);
}
}
```

→ prod input a1a2b3

output 1*2*3=6.

```
Public class product {
    Psvm (string args[]) {
String s = new string ("a1b2c9");
int product=1;
for(int x=0; x<s.length(); x++) {
    if(s.charAt(i)>='0' && s.charAt(i)<='9'){
        Product = product * character.get
NumericValue (s.charAt(i));
    }
    Syso (product);
}}
```

→ input:a3b2c1 output abbc.

```
Public class a3b2c1 {
  PSVM (String args[]){
  String s = new String ("a3b2c1");
  int num=
for(int x=0; x<s.length(); x++) {
if (s.CharAt(x)='a' && s.charAt(x)<='z'){
        temp=s.charAt(x);
    }
else if (s.charAt(x)>='0' && s.charAt(x)<='9'){
  num= Character.getNumeric Value(s.charAt(x));
for(int i=0; i<num; i++){
        Syso(temp);
    }
}
}
}
```

→ 2 strings 1 string is mutable and 1 string is immutable. accepting equals

```
String s = new String ("Hello").intern()
String s2 = "Hello";
if (s == s2) {
        syso("yes"); }
```

→ Replace:- new(stringBuffer(

```
String str = 'shot';
str.replace(0,4);
```

→ Delete:- Sb.delete(0,7)

Second largest word

```
Public class Secondbargeststring {
  PSVM (String args[]) {
  String s = new String("hello all good morning");
  String a[] = s.split(" ");
  String s3 = " ";
  int max=0, smax=0;
  for(int i=0; i<a.length;i++) {
      if (max<a[i].length()) {
        max=a[i].length();
      }
  }
  for (int i=0; i<a.length;i++){
    if (smax<a[i].length() && smax<max &&
    a[i].length()!=max) {
```

```
{ smax=a[i].length();
    s3=q[i];
  }
}
syso (s3+" "smax); }}
```

21/06/23 a13b12c11 → output should be printed 13 times

```
Public class a3b2c1 {
  PSVM (String args[]){
  String s= new String("a13b12c11");
    Char temp ='0';
  for (int x=0; x<s.length; x++){
        int num=0;
   if (s.charAt(x)>='0' && s.charAt(x)<='9'){
    num=Character.getNumericValue(s.charAt(x));
     num=num*10;
    if(s.charAt(x+1)>='0' && s.charAt(x+1)<='9')
    {
    int num1=Character.getNumericValue(s.charAt(x+1));
      num=num+num1;
       x++;
    for(int i=0;i<num;i++){
        syso(temp);
  }
 }
}
}
}
```

Q. Count the palindrome in a string

```
Public class Palindromsentence{
String s1 = new string ("madam arora teaches malayalam");
String s2[] = s1.split(" ");
    int i,j,k;
    String temp = "";
    String rev = "";
for (i=0; i<s2.length; i++){
        temp=s2[i];
        rev = "";
    for (j=temp.length()-1; j>=0;j--){
        rev=rev+temp.charAt(j);
    }
    if (rev.equals(temp)){
        Syso(temp); }}
```

→ Using StringBuffer.

```
→ for(k=0; k<s2.length; k++){
    StringBuffer sb=new StringBuffer(s2[k]);
    sb.reverse();
    String s4= sb.toString();
    if (s4.equals(s2[k])){
        Syso(s4);
    }
}
}
```

→ Input:- Madam Arora teaches malayalam;
Starting of every word should be capital.
MadamCapital{

```
String s1 = new String("hello all good morning");
string s2 [] = s1.split(" ");
String ans = " ";
for (int x=0; x<s2.length; x++){
        String s= s2[x];
ans = ans + s.substring(0,1).touppercase() +
s.substring(1, s.length()) + " ";
}
sysolans); }}
```

→ Input aaaabbbccc output a4b3c2.

```
Public class aaaabbbcc{
 Psvm(string args[])
 String s= new String("aaabbbccc");
 Char s2[] = s.tocharAnay();
 int count = 1, j;
boolean b[] = new boolean [s2.length];
for (int i=0; i<s2.length; i++){
        count=1;
      if(b[i]==true)
        continue;
   for(j=i+1; j<s2.length; j++){
     if (s2[i]==s2[j]){
          count++;
        b[j] = true; }}
  if (count>1){
    Syso(s2[i]+""+count);
}  }  }
```

Deletion of duplicate elements

```
Public class deletionduplicateelements {
    PSVM(String args[]) {
  String sb = new String Buffer("hello all good e
  morning");
     int x,y;
  for (x=0; x<sb.length(); x++){
    for (y=x+1; y<sb.length(); y++) {
      if (sb.charAt(x)==sb.charAt(y)) {
            sb.replace(y, y+1, "_");
          }}}
    String temp="";
  for(int i=0; i<sb.length(); i++) {
        if(sb.charAt(i)!=' '){
          temp = temp+sb.charAt(i);
        }
      }
      Sysoltemp);
  }
}
```

Unique elements in a String

```
Public class uniqueelements{
  Psvm (string[] args){
  String s1=new string("Hello all good morning");
  Char a[]=s1.tocharArray();
   int count=1,j;
  boolean b[]= new boolean [a.length];
for (int i=0; i<s1.length(); i++){
      if(a[i]!=' '){
          count=1;
       if (b[i]==true)
          Continue;
      for(j=i+1;j<s1.length();j++){
          if (a[i]==a[j]){
          count++;
          b[j]=true;
      }
unique  if(count==1){
          syso (a[i]+"-->"+count);}}}}
```

Duplicate

```
if (count>1){
    syso (a[i]+"-->"+count);}
```

Both duplicate and unique

```
syso (a[i]+" "+count);
```

Deletion of Duplicate elements / characters

```
String s1 = new String("Hello all good morning
char s2[]= s1.toCharArray(); → st
for(int i=0; i<s2.length; i++){
for(int j=0; j<s2.length; j++){
    if (s2[i] == s2[j]){
        break;
    }
}
if (i==j){
    ans=ans+s[i];
}
}
```

Deletion of Duplicate elements / Strings words.

```
Public class Duplicate_elements_words{
    Psvm (string args[]){ String s[]=s1.spilt(");
for(int i=0; i<s2.length; i++){
for(int j=0; j<s2.length; j++){
    if (s2[i].equals s2[j]){
        break;
    }
}
if (i==j){
String ans = ans + s2[i];
}
}
}
```

→ Lappy number.

192 → 192
192*2 → 384
192*3 → 576.

192384576

All the numbers from 1 to 9 should be their.

```
String s1 = new String("192");
int s2 = Integer.parseInt(s1)*2;
int s3 = Integer.parseInt(s1)*3;
String num = s1+s2+s3;   192384576
for(int i=0; i<num.length(); i++){
for(int j=i+1; j<num.length(); j++){
    if(num.charAt(i) == num.charAt(j)){
        break;
    }
}
if(i==j){
String ans = ans+num.charAt(i);
}
}  String s = "abcdefghijklmnopqrstuvwxyz";
if(ans.equals(s)){
    syso(ans);
}
```

The quick brown fox jumps over a lazy dog

```
Public class lappynumber{
  Public static void main(String args[]){
StringBuffer s1 = new StringBuffer("The quick brown fox jumps over a lazy dog");
String s3 = "abcdefghijklmnopqrstuvwxyz";
Char c2[] = s3.toCharArray();
Syso(c2);
in x,y;
  for(x=0; x<s1.length(); x++){
      if(s1.charAt(x)==s1.charAt(y)){
          s1.replace(y,y+1," ");
          }
      }
  String temp="";
  for(int i=0; i<s1.length();i++)
      {
      if(s1.charAt(i)!=' ')
      {
      temp=temp+s1.charAt(i);
  }
}
String s = String.valueOf(temp);
    s.trim();
  Char c[]=s.toCharArray();
    Arrays.sort(c);
    Syso(c) → if(Arrays.equals(c,c2)){
                Syso("Yes");}}}
```

Input Hello all Good Morning.
Output hELLO lla gooD gninroM

even → small to capital vice versa
odd → Reverse the words.

```
→ public class evenoddreverse {
   Psvm (string args[]){
    String s1 = new String ("Hello all Good Moming");
    String s2[] = s1.spilt(" ");
    String temp1 = " ";
    String ans = " ";
   for(int x =0; x<s2.length; x++) {
     if(x%2==0) {
        String temp = s2[x];
    for (int i=0; i<temp.length(); i++) {
      if (temp.charAt(i) >='A' && temp.charAt(i)<=
      {
       syso((char)(temp.charAt(i)+32)+" ");
      }
     else {
        syso((char)(temp.charAt(i)-32)+" ");
     }
    }
    else {
       temp1 = s2[x];
     for(int i= temp1.length()-1; i>=0; i--)
      {
       syso(temp1.charAt(i)+" ");
      }
   }
  }
}
```

Highest length in a string.

```
Public class highestlength {
    Psvm (string args[]) {
    string s = new string ("hello maheswari how
are you");
    String a[] = s.split();
     int x =0, i, index =0;
     for (i = 0; i <a.length; i++) {
         if (max <a[i].length()){
             index=i;
             max = a[i].length();
         }
     }
     Syso (a[index] +" " +max);

    }
}
```

Input :- Hello this is codinghub user
Output :- user codinghub is this Hello.

```
Public class replacingthestring {
    Public static void main (String args[]) {
    String s= new String ("Hello this is codinghub
user");
String a[]= s.spilt(" ");
    string temp=" "; temp1=" ";
     int i=0;
```

~~for~~

```
for (i=0; i<a.length; i++){
 if(a[i].equalsIgnoreCase(a[a.length-1])){
      temp = a[0];
       a[0] = a[a.length-1];
        a[a.length-1] = temp;
   }
  else{
    for(int j=a.length-1 ; j>=0 ;j--)
     {
      syso(a[j])); }} break; }}}
```

Sorting_without_arraymethod

```
Public class Sorting_without_arraymethods{
    Public static void main(String args[]){
     String s1 = new String("Apple");
     String s2 = "";
  for(int i=65; i<=122; i++){
  for (int j=0; j<s1.length(); j++){
     if((char)(i) == s1.charAt(j)){
       s2 = s2 + s1.charAt(j);
      }}}
      syso(s2); }}
```

Sorting words

```
Public class sortingstrings {
    PSVM (String args[]) {
    String s = new String("hello all good morning");
    String a[] = s.split(" ");
     String temp = a[0];
  for(int i=0; i<a.length-1; i++){
     if(a[i].length() > a[i+1].length()){
          temp = a[i];
          a[i] = a[i+1];
          a[i+1] = temp;
           i = -1
          }
    }
  for(String var : a){
        syso(var) + " "); } } }
```

Sorting chars

```
class characters {
    Public void checking(String s){
    char a[] = s.toCharArray();
     char temp = a[0];
  for(int i=0; i<a.length-1; i++){
       if(a[i]>a[i+1]){
            temp=a[i];
            a[i] = a[i+1];
            a[i+1]=temp;
        } }   i=-1;
    for(char var:a){ syso(var); } }
```

→ Character

Input :- 1100010001
Output :- 0011101110

```
Public class swap {
    Public static void main (String args[]){
        String s1 = new String("1100010001");
        char a[] = s1.toCharArray();
    for(int i=0; i<a.length; i++){
            if (a[i]=='1'){
                Syso('0');
        }
        else { syso('1'); } } }
```

→ Input :- Vital Information Resource Under Size"
Output :- VIRUS

```
Public class VIRUS {
    PSVM (String args[]){
String s1 = new String("vital Information
Resource Under size");
 String a[] = s1. split();
String s2 = " ";
Char ans = '0';
for(int i=0; i<a.length; i++){
        s2=a[i];
 for(int j=0; j<s2.length(); j++){
        ans = s2.charAt(0); }
    Syso(ans);
    }
}
```

→ Input :- Hello All Good MoRnInG
Output :- OLLLEh aLL DOOg moRNiNG

```
Public class hello{
   psvm (string args[]){
      String s= new String ("Hello ALL Good
   MoRnInG");
   String sb[]= s.split(" ");
    string s4 ="";
 for(int i=0;i<sb.length;i++){
      if(i%2==0){
      StringBuffer s1 =new StringBuffer(sb[i]);
         s1.reverse();
         s4=s1.tostring();
for(int j=0;j<s4.length();j++){
   if(s4.charAt(j)>='a' && s4.charAt(j)<='z')
    {
    Syso((char)(s4.charAt(j)-32)+" ");
    }
  else{
     Syso((char)(s4.charAt(j)+32)+" ");
     }
   }
 }
else{
   string s5=sb[i];
   for(int j=0;j<s5.length();j++)
    {
```

```
if (s5.charAt(j)>='a'  && s5.charAt(j)<='z')
  {
   syso ((char)(s5.charAt(j)-32)+" ");
  }
  else {
     Syso ((char)(s5.charAt(j)+32)+" ");
    }
  }
 }
}
}
```

Highest Vowels in a String.

```
Public class Vowels {
    Psvm (String args[]) {
     String s1="Happy New Year";
     String a=s1.split(" ");
    int max=0, index=0;
     int array[]= new int [a.length]; String s="";
for(int i=0;i<a.length;i++) { s=a[i];int count=0;
for(int j=0; j<s.length();j++){
    if(s.charAt(i)=='A' || s.charAt(i)=='I' || s.charAt(i)
='E' ||s.charAt(i)=='O' || s.charAt(i)=='U'){
      count++;
    }} array[i]=count;
   for(int k=0; k<array.length;k++){
       if(array[i]>max){ max=array[i];
                         index=i;}}}

     Syso(a[index]);
}  }
```