

psum (string) args

```

<
int res = lcm(30, 45);
cout ("res = " + res);
}

```

0 1 1 2 3 5 8 13 21 34
 n1 n2 n3
 n1 n2 n3

Fibonacci

class FibonacciSeries

```

public static void ps Fibbo (0,1);
ps (int n1, int n2)

```

```

{
    s.o.p ("n1", 0);
    s.o.p ("n2", 1);
    for (int i=0; i<8; i++)
    {
        n1 + n2 = n1+n2;
    }
}
```

1. b220) s.o.p ("n3", 1); → s.o.p ("n1", 1); → for single s.o.p call first

n1 = n2

n2 = n3;

}

psum [string] args

```

for {
    if (isFibbo (0,1));
    else
    {
        }
}

```

int n1=0;

int n2=1;

s.o.p (n1);

s.o.p (n2);

int count=0;

while (count<8)

{int n3=n1+n2;

s.o.p (n3);

Array programs

class program1

```

{
    psvar (string[] args)
    {
        int arr = new int[5];
        s.o.p ("size=" + arr.length);
        arr [2] = 1;
    }
}

```

arr [2] = 1;

psvar (string[] args)

arr [4];

("size=" + arr.length);

PAGE NO.

5

int arr2 = { 12, 47, 52, 6, 13, 9 };

1. S.O.P ("size = " + arr2.length);

(* int arr3;) error

arr3 = { 12, 47, 52, 6, 13, 9 }; * /

2. arr3.size = arr3.length; int arr4 = { 14, 52, 36 };

3. S.O.P ("size = " + arr4.length);

4. ?

5. Prop: class mindTree

int f :

public static int psFact (int num)

1. {

2. int oact = 1;

3. for (int i = num; i > 1; i--)

4. {

5. oact = oact * i;

6. }

7. return oact;

8. }

9. public static int testRec (int arr)

10. {

11. return (isFact (arr.length) - 1);

12. }

13. int arr[] = { 12, 6, 5, 10, 2 };

14. int res = testRec (arr);

15. S.O.P (res);

16. ?

1) WAP to determine missing elements in an array

17. ?

18. ?

19. ?

20. ?

21. M = { 12, 13, 18, 25, 30 }

22. j = 13

23. 13 < 16

24. 13

25. 19

26. 26

27. j = 14

28. 14 < 16

29. 14

30. 20

27

31. j = 15

32. 15 < 16

33. 21

28

34. 22

35. j = 16

36. 16 < 16 X

37. 23

29

38. 24

39. 25

40. 26

41. 27

42. 28

43. 29

44. 30

45. 31

q = 1

125

j = 17

1250

j = 18

1250

i = 19

19 = 20

j = 20

00 <

proj1:-

class missingnum

{

public static void

d

for (int i = 0;

1

for (int j = 0;

5

S.O.P (j);

32

psum (str);

2

int c3 = 0;

num (c3);

num (c3);

5

psum (str);

1

int i] arr = 8

9 = 1

arr [1] = 0

i = 2

arr [2] = 0

i = 3

arr [3] = 0

4

arr [4] = 0

5

arr [5] = 0

6

arr [6] = 0

7

arr [7] = 0

8

arr [8] = 0

9

arr [9] = 0

10

arr [10] = 0

11

ARUNS-	PAGE NO.	DATE
111	111	111

ARUNS

9 = 1 125

j = 17 12420

i = 18 12420

k = 19 19 < 20

j = 20 00 < 20

for (int i = 0; i < arr.length - 1; i++)

 i++

 for (int f = arr[i] + 1; f < arr[i + 1];

 f++)

 c.o.p(i);

 class MissingNumbers

 {

 public static void main (int [] ar)

 {

 for (int i = 0; i < arr.length - 1; i++)

 {

 for (int j = arr[i] + 1; j < arr[i + 1]; j++) // loop to displaying missing numbers

 c.o.p(j);

 }

 public sum (String [] args)

 {

 int [] arr = { 12, 16, 20, 22, 25 };

 num (arr);

 }

prob - write a program to reverse the given array element
int [] arr = { 1, 2, 3, 4, 5 };

i = 1 125

arr[1] = arr[3]

i = 2 215

arr[2] = arr[4 - 2]

i = 3 315

arr[3] = arr[4 - 3]

i = 4 415

arr[4] = arr[4 - 4]

arr

$\rightarrow \begin{array}{|c|c|c|c|c|} \hline 1 & 2 & 3 & 4 & 5 \\ \hline 0 & 1 & 2 & 3 & 4 \\ \hline \end{array}$

arr $\rightarrow \begin{array}{|c|c|c|c|c|} \hline 5 & 4 & 3 & 2 & 1 \\ \hline 0 & 1 & 2 & 3 & 4 \\ \hline \end{array}$

class ReverseArray

public static int [] rev (int [] arr)

int [] arr = new int [arr.length];

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

 arr[i] = arr[arr.length - 1 - i];

return arr;

DATE _____
PAGE NO. _____

ARUN'S

public sumC(strin g s) {
 int arr[] = {1, 2, 3, 4, 5};
 int curr[] = rev(arr);
 for (int i = 0; i < curr.length; i++) {
 d
 }

i = 0	arr[5]	arr[3] = 2	curr[5] =
int temp = 1	i = 2 arr[5]	temp = 3	$q = 0 \Rightarrow arr[0] \Rightarrow curr[5]$
arr[0] = arr[4 - 0]	temp = 3	arr[2] = arr[4 - 2]	$i = 1 \Rightarrow curr[1] \neq curr[3]$
curr[4] = 1	$\underbrace{arr[3]}_{arr[3] \neq arr[5]}$	$\underbrace{arr[2]}_{arr[2] \neq arr[3]}$	
i = 1	temp = 2	temp = 2	
arr[1] = arr[3]	arr[3] = arr[4 - 3]	arr[3]	

Q. class DoublyEndarry

1. public static int rev(int arr[])
 2. {
 3. int len = arr.length;
 4. for (int i = 0; i < len; i++) {
 5. curr[i] = arr[len - 1 - i];
 6. }
 7. return curr;
 8. }

ps. int rev(int arr[], int len) {
 for (int i = 0; i < len; i++) {
 }

for (int i = 0; i < arr.length - 1; i++) {
 curr[i] = arr[arr.length - 1 - i];
}

curr[i] = arr[arr.length - 1 - i];
 new curr
 new curr

return curr;

}

public static int rev(int arr[])
 {
 }

for (int i = 0; i < arr.length / 2; i++) {
 come oppy
 i
 arr[i] = arr[arr.length - 1 - i];
}

arr[arr.length - 1 - i] = temp;

temp = arr[i];

pcvn (string)
 1. int arr[] =
 2. int curr[] =
 for (int i =
 2. 5. o. e (no)
 2. 23

dryg: write a
 present in
 clues first
 1
 public static
 int maxSum
 int index =
 for (int i =
 5. 96 (max <
 1
 max = cur
 , index

5. 96 ("true"
 1. 0. 96 ("i"
 } psuml strin
 1
 int i = 0
 for (i = 0;
 }
 > Assign

1. cur =
 2. wap
 3. wap
 4. cur

program (string & int args)

4

int arr[] = {1, 2, 3, 4, 5};

int max[] = max2(arr);

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

2

System.out.println(max[i]);

223

Q3: write a program to determine first maximum element
present in the array along with its index.

class Firstmax

1

public static void First(int arr)

int max = arr[0];

int index = 0;

for (int i = 1; i < arr.length; i++)

3

if (max < arr[i])

4

max = arr[i];

index = i;

5

6

System.out.println("The maximum element is " + max);

System.out.println("It is present at " + index);

} (sum string to arrays)

7

int arr[] = {12, 41, 25, 3, 16};

First(arr);

8

> Assignments

1) write a program to determine first two max elements in an array

2) WAP to determine first three max number

3) WAP to determine nth max number in an given array, where n is decided by user.

21/8/19

i = 1 145

124

i = 2 215

12223

max2 = 123

max1 = 24

i = 3 225

2823

i = 4 225

28215

12215

Program to find 2nd max

int arr = {12, 3}

for (int i = 0; i < arr.length - 1; i++)

{

 for (int j = i + 1; j < arr.length; j++)

 if (arr[i] > arr[j])

 temp = arr[i];

 arr[i] = arr[j];

 arr[j] = temp;

 }

 }

 check here

class Sortarray

{

 public static void main(String[] args)

 {

 for (int i = 0; i < arr.length - 1; i++)

 for (int j = i + 1; j < arr.length; j++)

 if (arr[i] > arr[j])

 temp = arr[i];

 arr[i] = arr[j];

 arr[j] = temp;

 }

 }

 }

 return arr;
}

}

public class SecondMax

{

 int arr[] = {12, 4, 28, 3, 15};

 first = arr[0];

 second = arr[1];

 for (int i = 2; i < arr.length; i++)

 if (arr[i] > first)

 first = arr[i];

 else if (arr[i] > second)

 second = arr[i];

21/8/19

i=1 125

1224

1215

1228

max2 = 123

max1 = 24

i=3 325

2813

i=4 125

28215

1215

Ques: Write to sort the given array in ascending order

```
int arr = {12, 3, 18, 1, 9};
```

```
for (int i=0; i<arr.length; i++)
```

```
{
```

```
    for (int j=i+1; j<arr.length; j++)
```

```
{
```

```
        if (arr[i] > arr[j])
```

```
{
```

```
            int temp = arr[i];
```

```
            arr[i] = arr[j];
```

```
            arr[j] = temp;
```

```
}
```

arr - [1 3 18 12] i=2 225 i>1 1125 j=2 125

12 > 18 i=1 1125 j=2 125

j=3 325 temp=12 12 > 18

12 > 18 arr[2]=12 i=3 325

swap 12 > 12 3 > 4

arr[1]=12 12 > 12 3 > 4

arr[0]=12 12 > 12 3 > 4

class Sortarray

```
{
```

```
    public static int sort (int[] arr)
```

```
{
```

```
    for (int i=0; i<arr.length; i++)
```

```
{
```

```
        for (int j=i+1; j<arr.length; j++)
```

```
{
```

```
            if (arr[i] > arr[j])
```

```
{
```

```
                int temp = arr[i];
```

```
                arr[i] = arr[j];
```

```
                arr[j] = temp;
```

```
}
```

```
}
```

```
    }
```

```
}
```

```
PSVM (ction) (011)
```

```
{
```

```
int arr [] = {12, 3, 18, 1, 9};
```

```
int ans = sort (arr);
```

PAGE NO. _____ DATE _____

ARUN'S

```

for (int i=0; i<arr.length; i++)
{
    s.o.p(arr[i]);
}

```

Ques:- find max number
 Ans:- same as previous prg.
 Problem static int max (int arr[], int n)
 if (n > arr.length)
 return -1;

```

    {
        arr = sort (arr);
        return arr[arr.length - 1];
    }
}

```

Ques:- find max number
 Ans:-
 int arr[] = {12, 3, 8, 1, 4, 5};
 int a = 5;
 int res = max (arr, b);
 if (res == -1)
 l.o.p (res);
 else
 l.o.p ("invailed max number");
 }
}

Ques:- WAP to determine a no. minimum no. from a given array

Ans:- same as above

public static int min (int arr[], int n)

```

if (n > arr.length)
    return -1;
}

```

ARUN'S

PAGE NO.

DATE

Ans - [1] [3] [4] [12] [8]

1st max → 4 index

2nd max → 3 index

3rd max → 2 index

4th max → 1 index

5th max → 0 index

Ques:-
 arr = sort (arr);
 return arr[0];

Ans -

11 insert & on c

array

int arr[] = {1};

int ele = 25;

int index = 2;

int arr[2] =

arr[1] = 25;

arr[0] = 1;

arr[2] = 25;

arr[1] = 25;

for (int i=0;

i < arr.length;

arr[i] = arr[i+1];

}

arr[0] = 25;

int index = 3;

int element;

arr[3] = 25;

arr[0] = arr[0];

i = 1; i < 3;

arr[1] = arr[1];

return arr[0];

else

return arr[1];

else

return arr[2];

else

return arr[3];

else

return arr[4];

else

return arr[5];

else

return arr[6];

else

return arr[7];

else

return arr[8];

else

return arr[9];

Ans - [1] [3] [4] [12] [8]

3
arr[0] + (arr[i])

return arr[i-1];

function with iuuu datastructure
stack & queue

4

stack & queue

function in place

→ index

v → 3 index

arr → 2 index

arr → 1 index

new → 0 index

1) insert an element at specified position for the given array

int arr[] = {12, 4, 13, 2, 5};

int ele = 25;

int index = 2;

arr[i+index] = new int [arr.length+1];

arr[i+index] = ele;

for (int i=0; i< index; i++)

arr[i] = arr[i];

for (int i=index+1; i< arr.length; i++)

arr[i] = arr[i-1];

arr → [12 | 4 | 13 | 2 | 5]

arr → [12 | 4 | 25 | | |]

o r 2 0 * 5

arr[2] = 25

i = 0 < 2

arr[0] = arr[0];

i = 1 < 2

arr[1] = arr[0];

22/5/09

int index = 3

int element = 25

new[3] = 25

g>0 o<3

arr[0] = arr[0]);

i = 1 < 3

arr[1] = arr[1];

i = 2 < 3

arr[2] = arr[2];

i = 3 < 3 X

i = 3 < 5 X

arr[4] = arr[3];

i = 4 < 5

arr[5] = arr[4];

arr [1 | 2 | 3 | 4 | 5]

arr [1 | 2 | 3 | 25 | 4 | 5]

Program: Some utility scores & i

class Teacher marks

given
3/4/12/18
1
public static int insert(int arr[], int ele, int index)

if (index <= arr.length)

int arr[] = new int [arr.length+1];

arr[index] = ele;

for (int i=0; i< index; i++)

arr[i] = arr[i];

}

arr[2] = 25;

{}

ARUNA'S

DATE

```

for (int i = index; i < arr.length - 1)
{
    arr[i + 1] = arr[i];
}

return arr;
}

size
{
}

s.out("invalid index cannot perform the operation");
return arr;
}

int arr[] = {1, 2, 3, 4, 5};
int index = 2;

arr[0] = arr.length - 1;

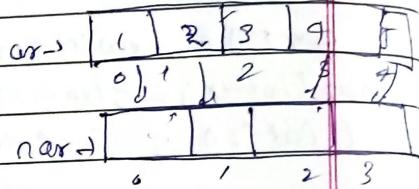
for (int i = 0; i < index; i++)
{
    arr[i] = arr[i];
}

arr[2] = arr[1];

for (int i = index + 1; i < arr.length; i++)
{
    arr[i] = arr[i];
}

return arr;
}

```



s.out("invalid index cannot occur as: ")

4

5

l → array name

n → size arr

a → array elem

insert

x → index

y → element

psum (method)

3

Scanner

read (* enter

int size = sc.nextInt();

int arr[] = new int[size];

for (int i = 0; i < size; i++)

 arr[i] = sc.nextInt();

 System.out.println("array is ");

 for (int i = 0; i < size; i++)

 System.out.print(arr[i] + " ");

 System.out.println();

 System.out.println("sum is ");

 int sum = 0;

 for (int i = 0; i < size; i++)

 sum += arr[i];

 System.out.println(sum);

 System.out.println("average is ");

 double average = sum / size;

 System.out.println(average);

 System.out.println("maximum is ");

 int max = arr[0];

 for (int i = 1; i < size; i++)

 if (arr[i] > max)

 max = arr[i];

 System.out.println(max);

 System.out.println("minimum is ");

 int min = arr[0];

 for (int i = 1; i < size; i++)

 if (arr[i] < min)

 min = arr[i];

 System.out.println(min);

ARUNA'S	DATE
3/3/2023	PAGE NO.

ARUNA'S	DATE
3/3/2023	PAGE NO.

{
s.o.p ("invalid index cannot perform the operation")
return 0;

}

l -> array name

n -> size array

a -> address

insert

x -> index

y -> element

5

12 0 1 2 8 12

2

insert

5 2 3

delete

0

psum (main (args))

{

Scanner sc = new Scanner (System.in)

s.o.p ("enter a array size")

int size = sc.nextInt () ; // size of the array

int arr [] = int [size] ;

for (i = 0 ; i < size ; i + +) // loop to take array elements

2

(o.s.p (arr)) s.o.p ("enter the element at + i + index")

String str = sc.next (); arr [i] = sc.nextInt ();

3

s.o.p ("enter the no of queries")

int q = sc.nextInt (); // no of queries

for (i = 0 ; i < q ; i + +) // next steps for queries

4

s.o.p ("entered the operation to perform")

String s1 = sc.next (); // operation to perform

if (s1 . equals ("insert"))

5

s.o.p ("enter the index") int si = sc.nextInt ();

int index = sc.nextInt ();

arr = delete (arr , index); arr = insert (arr , si , index);

6

else if (s1 . equals ("delete"))

7

enter the index;

DATE / /

PAGE NO. / /

ABUINNS

```

int index = sc.nextInt();
ar = delete(ar, index);
}
// end of function for loop
for (int i=0; i<ar.length; i++) {
    s.o.p(ar[i]);
}
}

// print like
1
2
3
Hello world
world

```

class ExampleMethod

```

{
public static void printArray (Object ar) {
    for (int i=0; i<ar.length; i++)
        s.o.p(ar[i]);
}
}

public static void main (String args) {
    Integer ar[] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
    printArray (ar);
    printArray (args);
    s.o.p ("Hello world!");
}

```

1) Int ar[] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
 ar = {1, 2, 3, 4, 5, 6, 7, 8, 9}

2) Print ar[2] {3, 3, 1, 2, 5, 3, 4, 5}

ARUN'S	PAGE NO.	DATE
--------	----------	------

1) wAP to delete
 n=6
 i=0 ok
 i=1 1<6
 if (ar[i] == ar[i+1])
 i>=2
 1<5
 ar[1] = ar[2];
 ar[2] = ar[3];
 3<5
 ar[3] = ar[4];
 4<5
 5<5
 2 > 1
 n=5
 i=0 ok
 i=1 1<6
 pr [0] = ar [2];
 2 == 2
 k=1
 while (k < 5)
 ar [1] = ar [2];
 k=2;
 ar [2] = ar [3];
 k=3
 ar [3] = ar [4];
 k=4
 ar [4] = ar [5];

1) wAP to delete Repeated element in an Array.

23/5/9

$n=6$	$i=6$	$arr[0]=08[0]$	$i=0$	$08[6]$
$i=0$ $08[6]$	$i=6$ $685x$	$arr[8]=08[8]$	$i=1$	186
$i=1$ 186	$i=1$	$arr[8] = arr[7]$	$i=2$	$08[8]$
$if (arr[0] == arr[1])$	$i=2$ 085	$arr[9] = arr[8]$	$i=2$	$2=2$
$8=8$	$if (arr[0] == arr[2])$	$i=1$	$i=1$	$1=5$
185	$2=3x$	$i=2$ $2=4$	$arr[7] = arr[2]$	
$arr[1] = arr[2]$	$i=3$ 325	$i=0$ $arr[7] = arr[2]$	$i=2$	
$08[5]$	$if (arr[0] == arr[3])$	$i=3$ $3=4$	$arr[2] = arr[3]$	
$arr[2] = arr[3];$	$2=2$	$i=3$ $3=4$	$08[3] = arr[4]$	
325	$k=3$	$i=0$ $arr[7] = arr[3]$	$08[3] = arr[4]$	
$arr[3] = arr[4];$	$3=4$	$i=1$	$arr[4] = arr[5];$	
415	$arr[3] = arr[4)$	$i=2$	$i=5$	
845	$k=4$ $+24$	323	$525x$	
	$n--$	$n--$		
	$n=4$	$n=3$		
		$i=4$ 423		
		$i=2$ $2C3$		
		$i=3$ 323		

22 7 1 1 3 2 5

$n=6$

$i=0$ 086

$j=1$ 186 {2 1 3 2}

$if (arr[0] == arr[1])$

$2 \geq 2$

$k=1$

while($k < 5$)

$arr[1] = arr[2] \rightarrow \{2\}$

$k=2$:

$arr[2] = arr[3] \leftarrow 2 \{3\}$

$k=3$ $\rightarrow \{1\}$

$arr[3] = arr[4] \rightarrow \{1\}$

$k=4$

$arr[4] = arr[5] \{1 3 2\}$

$\rightarrow \{2 3\}$

2 2 1 1 3 2
 n=6
 i=0 0<6
 j=1 1<6
 if ($i >= 2$)
 arr[1] = arr[2];
 arr[2] = arr[3];
 arr[3] = arr[4];
 arr[4] = arr[5];
 arr[5] = arr[6];

n=5 we missing 1 iteration
 j=2 2<5
 if ($arr[0] == arr[2]$)
 arr[0] = arr[3];
 arr[2] = arr[3];
 arr[3] = arr[4];
 arr[4] = arr[5];
 arr[5] = arr[6];

n=4
 j=5 5<4
 arr[1] = arr[2];
 arr[2] = arr[3];
 arr[3] = arr[4];

n=4
 j=1 1<4
 j=2 2<4 2 1 3
 if ($i == 1$)
 k=2
 2<3
 arr[2] = arr[3];

int arr = {2, 2, 2, 1, 3, 1};

n=6
 i=0 0<6
 j=1 1<6
 if ($i >= 2$)
 arr[1] = arr[2];

arr[0] 1 3 1
 arr[1] 2 3 4 5
 arr[2] one miss writer

arr[3] = arr[4];

arr[4] = arr[5];

arr[5] = arr[6];

arr[6] = arr[7];

ARUNA'S PAGE NO. 1575 DATE 15/5/23

class detectDuplicate
 + public (string arr){
 + int n = arr.length;
 + for (int i=0; i<n; i+)
 + for (int j=i+1; j<n; j+)
 + if (arr[i] == arr[j])
 + return true;
 + else
 + if logic for shift
 + if k=j;
 + (while (k<0-1))
 + k+;
 + arr[k] = arr[k+1];
 + k+;
 + } // end of while
 + n-; // ignore
 + j-; // next com
 + .
 + } // end of it
 + } // end of j
 + } // end of i
 + int arr = 0
 + for (int i=0; i<n; i+)
 + arr[i] = arr[i+1];

for (int i=0; i<n; i+)
 arr[i] = arr[i+1];

1 iteration

(less delete expected)

public static int[] dedupe(int[] arr)

2 1 1 3 2

-> 2 1 1 3

int n = arr.length; // length variable

for loop for comparison

for (int i = 0; i < n; i++)

for (int j = i + 1; j < n; j++)

if (arr[i] == arr[j]) // repeated element condition

1

// logic for shifting of element

if k = j;

while (k < n - 1)

1

arr[k] = arr[k + 1];

k++;

? // end of while

i--; // ignore last character b/c we shifted already

j--; // next comparison should happen from some shifted

* element *

? // end of it

? // end of j loop

? // end of i loop

int narr = new int[n];

for (int i = 0; i < n; i++)

{

narr[i] = arr[i];

1

return narr;

? // end of method

psum (String[] args)

{

int[] arr = {1, 2, 2, 2, 1, 3, 1};

int narr[] = dedupe(arr);

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

System.out.println(narr[i]);

PAGE NO.

DATE

ARRUN'S

Program to count freq of repeated element in an array

arr = {2, 2, 2, 3, 1, 3, 1, 2};
 arr = {1, 2, 2, 1, 2, 1};
 0 1 2 3 4 5

24(5/19)

i → is repeated address
 j → is repeated address
 int count = 0;

count = 0 n = 6

i = 0 0≤6

j = 1 1≤6

if (i == j) X

f = 2 2≤6

if (i == j) X

f = 3 3≤6

if (i == j) X

count = 1;

f = 4 4≤6

if (i == j) X

f = 5 5≤6

if (i == j) X

f = 6 6≤6

if (i == j) X

f = 7 7≤6

if (i == j) X

f = 8 8≤6

if (i == j) X

f = 9 9≤6

if (i == j) X

f = 10 10≤6

if (i == j) X

f = 11 11≤6

if (i == j) X

f = 12 12≤6

if (i == j) X

f = 13 13≤6

if (i == j) X

f = 14 14≤6

if (i == j) X

f = 15 15≤6

if (i == j) X

f = 16 16≤6

if (i == j) X

f = 17 17≤6

if (i == j) X

f = 18 18≤6

if (i == j) X

f = 19 19≤6

if (i == j) X

f = 20 20≤6

if (i == j) X

f = 21 21≤6

if (i == j) X

f = 22 22≤6

if (i == j) X

f = 23 23≤6

if (i == j) X

f = 24 24≤6

if (i == j) X

f = 25 25≤6

if (i == j) X

f = 26 26≤6

if (i == j) X

f = 27 27≤6

if (i == j) X

f = 28 28≤6

if (i == j) X

f = 29 29≤6

if (i == j) X

f = 30 30≤6

if (i == j) X

f = 31 31≤6

if (i == j) X

f = 32 32≤6

if (i == j) X

f = 33 33≤6

if (i == j) X

f = 34 34≤6

if (i == j) X

f = 35 35≤6

if (i == j) X

f = 36 36≤6

if (i == j) X

f = 37 37≤6

if (i == j) X

f = 38 38≤6

if (i == j) X

f = 39 39≤6

if (i == j) X

f = 40 40≤6

if (i == j) X

f = 41 41≤6

if (i == j) X

f = 42 42≤6

if (i == j) X

f = 43 43≤6

if (i == j) X

f = 44 44≤6

if (i == j) X

f = 45 45≤6

if (i == j) X

f = 46 46≤6

if (i == j) X

f = 47 47≤6

if (i == j) X

f = 48 48≤6

if (i == j) X

f = 49 49≤6

if (i == j) X

f = 50 50≤6

if (i == j) X

f = 51 51≤6

if (i == j) X

f = 52 52≤6

if (i == j) X

f = 53 53≤6

if (i == j) X

f = 54 54≤6

if (i == j) X

f = 55 55≤6

if (i == j) X

f = 56 56≤6

if (i == j) X

f = 57 57≤6

if (i == j) X

f = 58 58≤6

if (i == j) X

f = 59 59≤6

if (i == j) X

f = 60 60≤6

if (i == j) X

f = 61 61≤6

if (i == j) X

f = 62 62≤6

if (i == j) X

f = 63 63≤6

if (i == j) X

f = 64 64≤6

if (i == j) X

f = 65 65≤6

if (i == j) X

f = 66 66≤6

if (i == j) X

f = 67 67≤6

if (i == j) X

f = 68 68≤6

if (i == j) X

f = 69 69≤6

if (i == j) X

f = 70 70≤6

if (i == j) X

f = 71 71≤6

if (i == j) X

f = 72 72≤6

if (i == j) X

f = 73 73≤6

if (i == j) X

f = 74 74≤6

if (i == j) X

f = 75 75≤6

if (i == j) X

f = 76 76≤6

if (i == j) X

f = 77 77≤6

if (i == j) X

f = 78 78≤6

if (i == j) X

f = 79 79≤6

if (i == j) X

f = 80 80≤6

if (i == j) X

f = 81 81≤6

if (i == j) X

f = 82 82≤6

if (i == j) X

f = 83 83≤6

if (i == j) X

f = 84 84≤6

if (i == j) X

f = 85 85≤6

if (i == j) X

f = 86 86≤6

if (i == j) X

f = 87 87≤6

if (i == j) X

f = 88 88≤6

if (i == j) X

f = 89 89≤6

if (i == j) X

f = 90 90≤6

if (i == j) X

f = 91 91≤6

if (i == j) X

f = 92 92≤6

if (i == j) X

f = 93 93≤6

if (i == j) X

f = 94 94≤6

if (i == j) X

f = 95 95≤6

if (i == j) X

f = 96 96≤6

if (i == j) X

f = 97 97≤6

if (i == j) X

f = 98 98≤6

if (i == j) X

f = 99 99≤6

if (i == j) X

f = 100 100≤6

if (i == j) X

f = 101 101≤6

if (i == j) X

f = 102 102≤6

if (i == j) X

f = 103 103≤6

if (i == j) X

f = 104 104≤6

if (i == j) X

f = 105 105≤6

if (i == j) X

f = 106 106≤6

if (i == j) X

f = 107 107≤6

if (i == j) X

f = 108 108≤6

if (i == j) X

f = 109 109≤6

if (i == j) X

f = 110 110≤6

if (i == j) X

f = 111 111≤6

if (i == j) X

f = 112 112≤6

if (i == j) X

f = 113 113≤6

if (i == j) X

f = 114 114≤6

if (i == j) X

f = 115 115≤6

if (i == j) X

f = 116 116≤6

if (i == j) X

f = 117 117≤6

if (i == j) X

f = 118 118≤6

if (i == j) X

f = 119 119≤6

if (i == j) X

f = 120 120≤6

if (i == j) X

f = 121 121≤6

if (i == j) X

f = 122 122≤6

if (i == j) X

f = 123 123≤6

if (i == j) X

f = 124 124≤6

if (i == j) X

count in arr

24/5/19

repeated (cont repeated)

1 public static void counter (int arr)

2 int n = arr.length; // length variable

3 for (i loop for comparison)

4 for (int i=0; i < n; i++)

5 }

6 int count = 0;

7 for (int i=0; i < n; i++)

8 }

9 if (arr[i] == arr[i+1]) // repeated element condition

10 }

11 count++;

12 } logic for shifting of elements

13 int k = i;

14 while (k < n-1)

15 }

16 arr[k] = arr[k+1];

17 k++;

18 } // end of while

19 --i; // ignore last character

20 --i // next comparison should happen from some shifted element

21 end of if loop

22 end of i loop

23 if (arr[i] + " is repeated" + count + " times")

24 end of i loop

25 end of method

26 return count;

27 }

28 int arr = {1, 2, 2, 1, 3, 1};

29 count (arr);

30 }

31 }

WAP to determine the sum of non-repeated elements in an array.

class countRepeating

{

public static void main.

```
int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

```
int sum = 0;
```

```
for (int i=0; i<n; i++)
```

{

```
int count = 0;
```

```
for (int j=i+1; j<n; j++)
```

{

```
if (arr[i] == arr[j])
```

{

```
count++;
```

```
int k=j;
```

```
while (k<n-1)
```

{

```
arr[k] = arr[k+1];
```

```
k++;
```

```
i--;
```

```
j--;
```

}

}

```
if (count == 0)
```

{

```
sum = sum + arr[i];
```

}

}

```
System.out.println sum;
```

}

* public static void main (- - -)

{

```
int arr[] = {12, 12, 24, 1, 3, 12};
```

```
int sum = countRepeating(arr);
```

```
System.out.println sum;
```

WAP to calculate

int countRepeating {

int c;

2 m

class countRepeating

{

public static void

{

```
int a = arr.length;
```

```
for (int i = 0; i <
```

{

```
int count = 1;
```

```
for (int j = i + 1; j <
```

{

```
if (arr[i] == arr[j])
```

{

```
count++;
```

```
int k = j;
```

```
while (k < n - 1)
```

{

```
arr[k] = arr[k + 1];
```

```
k++;
```

```
i++;
```

```
j++;
```

```
}
```

```
if (count == 1)
```

```
sum += arr[i];
```

```
}
```

```
}
```

```
System.out.println sum;
```

```
}
```

```
return sum;
```

}

```
int sum = countRepeating(arr);
```

System.out.println sum;

}

DATE

PAGE NO.

ARUN'S

left consecutive.

int count[2] = {1, 1, 1, 2, 1, 2, 3};

1st occurs 3 times

2nd occurs 1 time

1st occurs 1 time

2nd occurs 1 times

class Counter

public static void Counter (int arr)

int a = arr.length; // length -> bubble sort so in parallel part

for (int i = 0; i < n; i++)

{

int count = 1;

for (int j = i + 1; j < n; j++)

{

if (arr[i] == arr[j])

{

count++;

else k = j;

cout << arr[i - 1].

}

arr[i] = arr[k + 1];

k++

{

i++;

}

}

cout << "Total " << count << endl;

}

sum (string & any)

{

int arr[] = {1, 1, 1, 2, 1, 2, 2, 3};

Counter (arr);

}

1) $i=0 \rightarrow "ababcaad"$

o $f \rightarrow "abcaad"$

2) $i=1 \rightarrow "acabcaad"$

o $f \rightarrow a \rightarrow 3$ times

b $\rightarrow 1$ time

c $\rightarrow 2$ times

d $\rightarrow 1$ time

3) $i=2 \rightarrow "aceccbaacc"$

o $f \rightarrow abcdacc$

Assignment

1) W.A.P matrix 90° length

2) W.A.P matrix to rotate right 90° longer

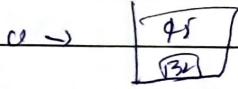
3) W.A.P. \rightarrow π \rightarrow 180° longer

4) W.A.P. \rightarrow π \rightarrow 90° left

2D - arrays

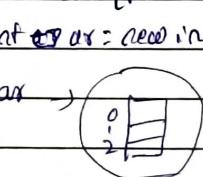
Data type v.n = Value;

in ar = +5;



Data type (int arr);

int arr = new int[5];

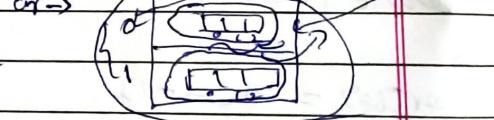


Data type (int arr) arr = new Data type();

int size; int arr[5];

int arr[5] = new int[5];

arr = new int[5];



Q7 | S | 14

class clockwise

)

return (string)

pat[0][0] or

pat count = 1;

for (int start =

2

for (int p =

1

arr[stor +] [i]

)

total point =

2

or (p)

)

for (int p =

2

arr[stor +]

)

for (int i =

2

arr[stor +]

)

ARUNS

ex1:- class program

{
 public (String s) {
 }

// 2D array

Ext [] [] arr = new Ext[3][3];

arr [0] [2] = 05;

arr [1] [0] = 40;

S.0.P ("Outer array size=" + arr.length); \Rightarrow 2

S.0.P (arr[0].length); \Rightarrow 3

S.0.P (arr[1].length); \Rightarrow 3

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

d

for (int j=0; j < arr[i].length; j++)

2

0025

4000

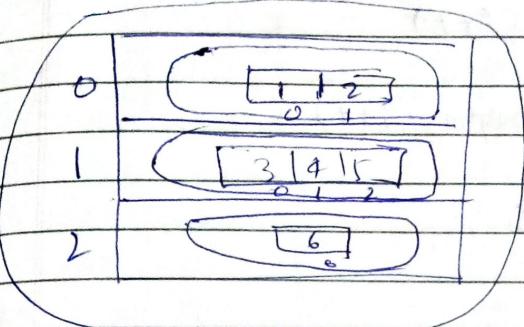
DATA	0	1	2	3
PAGE NO.	5.0	5.1	5.2	5.3
RENS	5.4	5.5	5.6	5.7

without need of operator.

int arr = {1, 2, 3} → single

int arr = {{1, 2}, {3, 4, 5}, {6}};

1) occurring several
repeated element



2) wap to display the numbers in clockwise direction

for

occurs clockwise direction

87 | 5 | 14

int arr = new datatype

arr[0] = 87; arr[1] = 5; arr[2] = 14;

using (string > arr)

int arr[3] are new arr[0] = 87;

arr[1] = 5; arr[2] = 14;

for (int start = 0; end > arr.length - 1; start++, end--) start++; end--;

2

for (int p = start; p < end; i++)

start → p

1

arr[start][i] = count++;

)

for (int i = start; i < end; i++)

2

arr[i][end] = count++;

)

for (int i = end; i > start; i--)

2

arr[end][i] = count++;

)

for (int i = end; i > start; i--)

)

arr[i][start] = count++;

)

1	2	3	4	5
16	17	18	19	6
10	11	12	13	14
15	16	20	21	7
80	51	25	23	24
19	23	22	21	8
30	31	32	33	34
13	12	11	10	9
40	41	42	43	44

```

if (arr.length != 0)
    arr[0] = arr[1];
    count = 1;
    for (int i=0; i<arr.length; i++)
        for (int j=0; j<arr[i].length; j++)
            System.out.print(arr[i][j] + " ");
    System.out.println();
}

```

QAP to determine

1	2	3
6	0	02
4	5	6
10	u	12
2	8	9
90	21	22

i = 0; j = 3
i = 0; 0 2 3
arr[0][j] = arr[0][i];
i = 1 1 2 3
arr[1][j] = arr[1][0];
j = 2 2 2 3
arr[2][j] = arr[2][0];

3) QAP to display the numeric in Anti-diagonal wise direction.

1	16	15	14	13
2	17	24	23	12
3	18	25	22	11
4	19	20	21	10
5	6	7	8	9

expt:
: Using transpose matrix
public static int transpose
int arr[2][2] arr =
for (int i=0; i<arr.length; i++)
 for (int j=0; j<arr[i].length; j++)
 arr[i][j] = arr[j][i];
 return arr;
}

int arr[2][2] arr =
System.out.println("Anti-diagonal elements are : ");
for (int i=0; i<arr.length; i++)
 for (int j=0; j<arr[i].length; j++)
 System.out.print(arr[i][j] + " ");
 System.out.println();
 return arr;
}

28/5/19

QAP to determine transpose of given matrix

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \end{bmatrix}$$

int C3C3 cur = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12};
 int r3C7 narr = new int[C3C3[2]];
 for (int i=0; i<narr.length; i++)
 for (int j=0; j<narr.length; j++)
 narr[i][j] = cur[i][j];

$$\begin{array}{l} i=0, j=3 \\ i=0, j=3 \end{array} \quad \begin{array}{l} i=3 \\ j=3 \end{array} \quad \begin{array}{l} i=1 \\ j=3 \end{array}$$

$$narr[0][i] = ox(i)[i]; \quad i=0, j=3 \quad \checkmark$$

$$i=1, j=3 \quad \checkmark$$

$$ox[1][0] = ox[0][1]$$

$$i=2, j=3$$

$$ox[2][0] = ox[0][2] = 7$$

you do

Code & directly

prob:

(QAP transpose matrix)

public static int C3C7 transpose (int C3C3 or)

int r3C7 [] narr = new int[narr.length] [or.length];
 for (int i=0; i<narr.length; i++)

for (int j=0; j<or.length; j++)

$$or[0][j] = or[0][j]$$

}

return narr;

}

PSJm (shubh) for arrays

}

int C3C7 [] cur = {{1, 2, 3, 4}, {5, 6, 7, 8}};

S-O-P // for input array is ..");

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

for (int j=0; j<or[i].length; j++)

}

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 5 \\ 2 & 6 \\ 3 & 7 \\ 4 & 8 \end{bmatrix}$$

ARUNI'S	PAGE NO.	DATE
5.0.8	8/12/2018	8/5/19
S-O-P ()		

Q) code to reverse the elements present in the row in a 2D matrix

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

i = 0 o[4]

j = 0 o[2]

temp = o[0][0] = 1;

o[0][0] = o[4][1] = o[0][1]

= o[0][3];

o[0][3] = 1;

i = 1 o[2]

temp = o[0][1] = 2;

o[0][1] = o[4][1-1]

= o[0][2];

o[0][2] = 2;

j = 2 o[2]

o[0][2] = 2;

class rowElementsReverse

public static int rowElementsReverse(int arr[])

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

for (int j=0; j<arr[i].length; j++)

int temp = arr[i][j];

arr[i][j] = arr[i][arr.length-1-j];

arr[i][arr.length-1-j] = temp;

}

}

return arr;

}

public static void main()

{

int arr[][] = {{1,2,3,4}, {5,6,7,8}, {9,7,8,5}, {9,7,3,12}};

S.O.P. arr

correctly it is - 4 |

for (int i=0;

for (int j=0;

{

S.O.P (arr)

}

S.O.P ());

2) (0) (clear)

public static

{

for (int i=0;

{

for (int i=0;

{

int temp =

arr[i][j] = arr[i][arr.length-1];

arr[i][arr.length-1] = temp;

}

return arr;

}

in main me

int arr[] =

S.O.P (arr);

for (int i=0;

{

for (int i=0;

{

for (int i=0;

{

S.O.P

}

S.O.P (arr);

}

return arr;

}

S.O.P (arr);

}

S.O.P ());

}

ARUNS

ARUNS

PAGE NO.

ARUNS

question

```
for (int i=0; i<arr.length; i++)
```

```
    for (int j=0; j<arr[i].length; j++)
```

```
        s.o.p("arr[" + i + "][" + j + "]:");
```

```
s.o.p();
```

2) col cleaver never Scan all arr org connige
public static int colcleaver (int arr[])

```
for (int i=0; i<arr.length/2; i++)
```

```
    for (int j=0; j<arr[i].length; j++)
```

```
        int temp = arr[i][j];
```

```
        arr[i][j] = arr[arr.length-1-i][j];
```

```
        arr[arr.length-1-i][j] = temp;
```

```
}
```

```
return arr;
```

```
}
```

In main method

```
int arr[] = colcleaver (arr);
```

arr tree col elements arranged correctly

```
for (int i=0; i<arr.length; i++)
```

```
{
```

```
    for (int j=0; j<arr[i].length; j++)
```

```
{
```

```
        s.o.p("arr[" + i + "][" + j + "]:");
```

```
}
```

```
s.o.p();
```

```
{} {}
```

1/ matrix multiplication

$$\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix} \Rightarrow \begin{matrix} 1 & 4 & 2 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{matrix} \Rightarrow \begin{matrix} 3 & 8 & 9 \\ 2 & 5 & 8 \\ 1 & 4 & 7 \end{matrix}$$

$$\begin{matrix} 3 & 6 & 9 \\ 2 & 5 & 8 \\ 1 & 4 & 7 \end{matrix}$$

te

car

4

in

18

11

a

11

c

6

3

1

p

s

i

v

89
58
42

ex:-

java is easy

30/5/19

class ReverseString

easy is java

```

public static String revl(String s1)
    String ocat = " ";no space → char ch[i] = s1. toCharAt(i);
    for (int i=0; i<ch.length; i++)
    {
        String res = "";no space
        while (i < ch.length & ch[i] != ' ')
            res = res + ch[i];
        i++;
        ocat = res + " " + ocat;
    }
    return ocat;
}

```

o) sum (String args)

```

String ocat = revl("java is easy");
S.O.P ("out" + ocat);
}

```

prg:- class ReverseString

i/o -> java. is easy

java + is easy

public static String revl(String s1)

class { ch=s1. toCharAt(i);

String ocat = " ";^{no space}

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

^{no space.}

String res = "";

while (i < ch.length & ch[i] != ' ')

}

res = res + ch[i];

i++;

ocat = res + ocat + " " + res. length();

PAGE NO.

ARUN'S

return ocat;

sum (String s)

{

String static

String ocat = "

S.O.P ("out" +

{

}

3) CAR to CO

class convention

{

static String

"seven", "eight"

"fifteen", "six

static String

"sixten", "seventy

public static

ig (num)

{

C.O.P (second)

{

else

{

S.O.P (first)

{

if (num >

C.O.P ("S1")

{

public static

{

int a=0;

if (aem =

{

int a=0;

if (aem =

{

S.O.P (

1 / 1

ARUN'S

30/5/19

psum(string args)

}

String static void main(

String out = new ("Hello World") ;

s.o.p ("out=" + out);

}

}

3) convert to convert number into words.

class ConvertWord

{

static String first = {"one", "two", "three", "four", "five", "six",
"seven", "eight", "nine", "ten", "eleven", "twelve", "thirteen", "fourteen",
"fifteen", "sixteen", "seventeen", "eighteen", "nineteen"};

static String second = {"", "", "twenty", "thirty", "forty", "fifty",
"sixty", "seventy", "eighty", "ninety"};

public static void print (int num, String s1)

{

if (num > 19)

{

s.o.p (second [num / 10] + " " + first [num % 10]);

}

else

{

s.o.p (first [num]);

}

if (num > 0)

s.o.p ("");

{

public static void main (String [] args)

{

int num = 123456789;

if (num == 0)

{

s.o.p ("zero");

size

5

```
print (num / 100000000, "crore");
print ((num / 100000) % 100, "lakhs");
print ((num / 1000) % 100, "thousand");
printf ((num / 100) % 10, " hundred ");
printf ("%d", num % 100);
```

)

4) WAP to determine count the no. of words in a given

string

```
if (str == " ") return 0;
int count = 1;
```

```
for (int i=0; i<str.length(); i++)
```

{

```
if (str[i] == ' ' && str[i+1] != ' ')
```

2

```
count++;
```

}

```
return count;
```

class Counter

{

```
public static int counter (String s)
```

2

```
char ch[] = s.toCharArray ();
```

```
int count = 0;
```

```
if (ch[0] == ' ')
```

```
count = 0;
```

```
for (int i=0; i<ch.length-1; i++)
```

{

```
if (ch[i] == ' ' && ch[i+1] != ' ')
```

}

```
count++;
```

return count;

}

return loc

3

PSUM (str)

2

int res = 0;

s.o.i l

2

2

string res =

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

2

if (cnt == 0)

res = s

2

return res;

class Ano

public static int

char E[] =

string re

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

2

if (ch[i] == ' ')

res = "

return

5

public sta

char ch[] =

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

{

if (ch[i] == ' ')

count++;

return count;

5

class C

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

{

if (ch[i] == ' ')

count++;

return count;

PAGE NO.

ARUNA'S

DATE

return count;

}

psum (string & vars)

2

int res = counter (" Java is a programming language ");
cout < "res = " + res;

3

Anagram

string res = "";

for (int p=0; p < ch.length(); p++)

2

if (cn[i] != 1)

res = res + cn[i];

}

return res;

(ans Anagram

public

static string removeSpace (string s1)

1

char [] ch = s1. tolowercase ();

string res = " ";

for (int p=0; p < ch.length(); p++)

2

if (ch[i] != " ")

res = res + ch [i];

3

return res;

4

public static string toUppercae (string s1)

1

char ch [] = s1. tolowercase ();

for (int p=0; p < ch.length(); p++)

2

if (ch[i] == 'a' + p + ch [i] - 'A')

ch [i] = (char) (ch [i] - 32);

ANS
PAGE NO.

return needString(ch);

} public static String sort(String s)

{ char c[] = s.toCharArray();

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

{ for (int j=i+1; j < ch.length; j++)

{ if (ch[i] > ch[j])

{ char temp = ch[i];

ch[i] = ch[j];

ch[j] = temp;

}

}

return needString(ch);

} public static boolean compare(String s1, String s2)

{ char c1[] = s1.toCharArray();

char c2[] = s2.toCharArray();

for (int i=0; i < c1.length; i++)

{ if (c1[i] != c2[i])

return false;

}

return true;

}

} public void (String s) { args}

{ string s1 = "Mother in Law";

string s2 = " Miller Woman";

C.O.P ("true given input string");

C.O.P (" " + s1);

S.O.P (" " +

s1 = "mene

s2 = "men

S.O.P (" " +

S.O.P (" " + s1 +

C.O.P (" " + s2 +

, s1 + s1.length());

2

s1 = "Dupper

s2 = "top

S.O.P (" " +

S.O.P (" " + s1 +

S.O.P (" " + s2 +

s1 = "co

s2 =

C.O.P (" " + s1 +

C.O.P (" " + s2 +

holean

if (true

S.O.P ("

else

S.O.P ("

)

else

1

S.O.P

2

?

1

O/P -

→

else

S.O.P ("S₂ = " + S₂);

S₁ = removespace(S₁);

S₂ = removespace(S₂);

S.O.P ("the string after removing space");

S.O.P ("S₁ = " + S₁)

S.O.P ("S₂ = " + S₂)

If (S₁.length() == S₂.length())

2

S₁ = tolowercase(S₁);

S₂ = tolowercase(S₂);

S.O.P ("the after converting to one case");

S.O.P ("S₁ = " + S₁)

S.O.P ("S₂ = " + S₂)

S₁ = sort(S₁);

S₂ = sort(S₂);

S.O.P ("the string after sort after sorting are");

S.O.P ("S₁ = " + S₁)

S.O.P ("S₂ = " + S₂)

boolean res = compare(S₁, S₂);

If (res)

S.O.P ("strings are equal");

else

S.O.P ("strings are not equal");

3

S.O.P ("strings are not equal");

4

S.O.P ("strings are not equal");

5

I/O → computer COMPUTER

→ memory PROGRAM

After create

String res = "";

```
for (int i=0; i<c1.length(); i++)  
{  
    for (int j=0; j<c2.length(); j++)  
    {  
        if (c1[i] == c2[j])  
            res = res + c1[i];  
    }  
}  
return res;
```

i=0 o/p

j=0 o/p

c1[i] = c2[j]

'c' i = 3

class stringModify

```
public static String unique(String s1, String s2)  
{
```

char c1 = s1.toCharArray();

char c2 = s2.toCharArray();

String res = "";

```
for (int i=0; i<c1.length(); i++)  
{
```

int counter = 0;

```
for (int j=0; j<c2.length(); j++)  
{
```

if (c1[i] == c2[j])
{

counter++;

}

if (counter == 0)

res = res + c1[i];

}

return res;

public (String s1, String s2)

{

String s1 = "ca"

String s2 = "ba"

String res = un

concp("res = " +

");

}

System.out.println(res);

String s1 = "ca"

String s2 = "ba"

String res = un

concp("res = " +

");

PrintWriter

out.println(res);

}

i++

j++

i=0 o o

j=0 k=

while (o < 12 & o <= 12)

j == 12

o == 12

i = 1 1a12

k = 1 j = 0

o == v

o == 12

class subst

{

public (String str,

{

char c1

char c2

String str1

String str2

String res = str1

for (int i=0; i<str1.length(); i++)

if (str1[i] == c1)

res = res + c2;

else

res = res + str1[i];

return res;

}

}

pxm (string \Rightarrow arrge)

1

String S1 = " computer .."

String S2 = " program";

String arr = unique (S1, S2);

C.P.P ("mes" + mes);

}

}

substring

1 | 6 | 14

String S1 "iduv id scdy" String S2 = "VV".

char C1 = S1. to charArray (C1);

i = 0

o < 12

i = j = 2

char C2 = S2. to charArray (C2);

j = 0

C.P.C1 = C2

for (int i = 0; i < C1.length(); i++)

'i' = 'V'

}

i = 1 o < 12

Pnt i = 0;

i = 0

while (C1[i] == C2[j])

'i' = 'V'

<

i = 1

i++;

j = 0

j++;

'V' = 'V'

i = 0 o < 12

'i' = 'V'

j = 0 k = 0

i = 0

while (o < 12 & o < 2 & C1[o] == C2[o])

j = 'V'

f = C2

o = -2

i = 2 o < 12

V > 2 j > 0

i = 1 o < 12 V >= V

k = 1 j = 0 k = 3 j = 1

U = V o = -1 k = 4 j > 2

o = -2 2 > 2

return true

class substring

1.

public static boolean sub (String S1, String S2)

}

char C1 = S1. to charArray (C1);

char C2 = S2. to charArray (C2);

DATE PAGE NO. ARUNA'S

for (int p=0; p < c1.length(); p++)

8

int j=0;

9 int k=i;

while (k < c1.length() && j < c2.length() && c1[k] == c2[j])

10

k++;

j++;

11

if (j == c2.length())

12

return true;

13

14

return false;

15

return (string(s) == "

16

Boolean res = substr("java is really , "ver");

17 if (res)

18

19 cout << " it's a substring";

20

21 else

22

23 cout << " it's not a substring";

24

repeated substring

clues substring

25

particular string part substr(string s1, string s2)

26

char c1 (1=s1. to chararray());

char c2 (2=s2. to chararray());

last count = 0;

for (int p=0; p < c1.length(); p++)

1

int j=0;

2 int k=i;

3 while (k < c1.length() && j < c2.length() && c1[k] == c2[j])

4

5 k++;

6 j++;

7

8 if (j == c2.length())

9

10 count++;

11

12 return count;

13

14 result (string(s) == "

15

16 cout << substr(s, 0, i);

17

18 if (res == substr(s, 0, i))

19

20 cout << " it's a substring";

21

22 else

23

24 cout << " it's not a substring";

25

26

27

28

29

30

31

32

33

34

35

for (int i=0; i<c1.length; i++)

1

int j=0;

int k=i;

while (k < c1.length & & j < c2.length & c1[k] == c2[j])

}

k++;

j++;

2

if (j == c2.length)

}

count++;

}

)

return count;

3

result(spinning argc)

4

int res = strcmp("luvlu-luv-lava", "lva");

l.o.p ("res" "res");

5

l h

5 9 1 7 2 7 1 2 2 4

h → no. of lines

2 l d

1 12

4 → position no

≡

4 3 7 3 4 3 6 5 2

v → line no

3) h = 5

6

4 → queries

x y

3 20 22 33

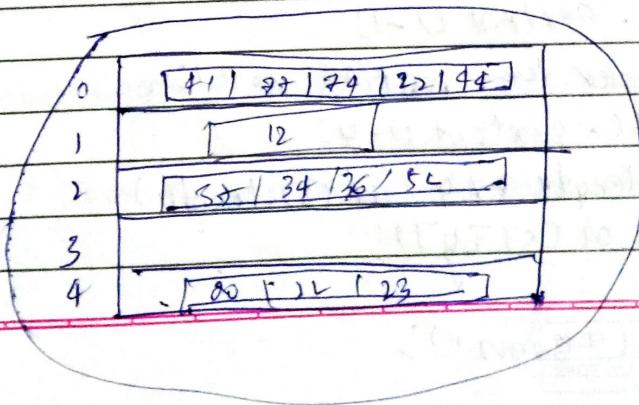
1 3 => 74

3 1 => 37

55 => Error

3 4 => 52

4 3 => Error



import java.util.Scanner;

class TeacherBank

{

public static void main (String args)

{

Scanner sc = new Scanner (System. in);

s. o. p ("Enter the no of lines");

int n = sc. nextInt();

int arr[] = new int [n];

for (int k = 0; k < n; k++)

{

s. o. p ("Enter the size of the "+(k+1)+" array");

int d = sc. nextInt();

int d = sc. nextInt(); int curr = new int [d];

arr[k] = curr; for (int i = 0; i < d; i++)

{

s. o. p ("Enter the no of queries");

int q = sc. nextInt();

for (int j = 0; j < q; j++)

{

s. o. p ("Enter the value at "+q+" position");

curr[q] = sc. nextInt();

{

curr[j] = curr;

{

s. o. p ("Enter the no of queries");

int a = sc. nextInt();

for (int i = 0; i < a; i++)

{

s. o. p ("Enter the line no");

int x = sc. nextInt() - 1;

s. o. p ("Enter the position no");

int y = sc. nextInt() - 1;

if (x > curr.length || y > curr[x].length)

s. o. p ("Array Index Out Of Bound");

size

31
ARUNS
PAGE NO.

String s1 = "joe";

s string [] str =

str[0] iouu

1 is

2 Easy

String s1 = "joe";

s2 = "

char ch = s1

int temp = 0

String [] str =

for (int i = 0

{

String (ch)

{

res = res + ch

if (

{

str [temp] =

>

class split

{

public static

{

cur ch

int c

if (ch

count

for (int i = 0

{

if (ch == 7

{

String s = "Java is easy";

3/5/19

s string [] str = s.split (" ");

("a").

str[0] Java

1

2 Easy

str[0]

1

2 is e

3 ay

String s1 = "Java is easy";

s2 = " ";

char [] ch = s1.toCharArray();

int temp = 0;

String [] str = new String [counter (s1)];

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

{

String ch[i] != ' '

}

res = res + ch[i];

if (

)

str [temp++] = res;

>

class split Demo

{

public static int counter (String s1, String s2)

{

char ch[] = s1.toCharArray ();

int count = 1;

if (ch[0] == s2.charAt(0))

count = 0;

for (int i=0; i < ch.length - 1; i++)

{

if (ch[i] == s2.charAt(0) & ch[i+1] == s2.charAt(0))

{

count ++;

return count;

DATE	/	/
PAGE NO.	1	1
ARULIN'S		

public static string[] split (string s1, string s2)

string str2 = new string [counter(s1, s2)];
char ch[] = s1. ToCharArray ();

int temp = 0;
for (int i=0; i<ch.length; i++)

String res = "";
while (i < ch.length & ch[i] != s2.charAt(0))

{
res = res + ch[i];

i++;
if (temp == 0) // initial program we could only
return word

str [temp] = res;

}

return str;

}

psum (String [] arrs)

{
String s1 = "This is code";

String [] str = splits (s1, "i");
for (int p=0; p < str.length; p++)

{

System.out.println (str[p]);

}

}

Output:-

class StringDemo

{

psum (String [] arrs)

{

String s1 = "Java is a programming language";

String str [] = s1.split (" ");

String out = "";

(call class.)

for (int i=0;

i < str.length;

out = out + str[i];

out = out + " "

out = out + str[i];

out = out + str[i];

System.out.println (out);

}

}

IP :- " Java

" Java

class String Demo

{

public static

char ch [] =

String out = "

for (int p=0;

p < str.length;

{

String res = str[p];

out = out + res;

}

System.out.println (out);

}

if (true)

out = out + ch[p];

}

52/

```
for (int i=0; i<str.length(); i++)
```

}

```
if (ix2 == 0)
```

```
out = out + str[i] + " ",
```

else

```
out = out + string reverse(str[0:i]) + " ",
```

else

```
cout << out;
```

}

}

```
IP :- Java is a programming language  
"Java is a programming
```

```
class string demo3
```

5

```
public static string firstcharupper(string s,)
```

{

```
char ch[2] = s[0].toCharArray();
```

```
string res = " ";
```

```
for (int i=0; i<ch.length(); i++)
```

{

```
string res = searchres
```

```
res += (i == 0) ? ch[0] : ch[i];
```

}

```
if (res.length() == 1)
```

2

```
if (ch[i] >= 97 & ch[i] <= 122)
```

```
ch[i] = (char) (ch[i] - 32);
```

3

```
res = res + ch[i];
```

```
++i
```

4

```
out = out + res + " ";
```

5

```
return out;
```

}

psum (string > args)

```
{  
    string s1 = "java is a programming language";  
    string res = firstCharUpper(s1);  
    s.o.p ("res = " + res);  
}
```

3) write to remove the vowels from the given string.

class removeVowels

```
{  
    public static string removeVowels (string s1)  
    {
```

```
        char [] ch = s1.toCharArray();
```

```
        string oct = "";
```

```
        for (int i = 0; i < ch.length; i++)
```

```
        {  
            if (ch[i] == 'a' || ch[i] == 'e' || ch[i] == 'i'  
                || ch[i] == 'o' || ch[i] == 'u')  
            }
```

```
}
```

```
}
```

```
    return oct;  
}
```

```
} psuon (main?? args)
```

```
string res = removeVowels ("edocorner");
```

```
s.o.p ("res = " + res);
```

```
}
```