NOTE: A prime number is a number which is only divisible by 1 and itself.
The first ten prime numbers are 2, 3, 5, 7, 11, 13, 17, 19, 23 and 29.

Task 15

Way 01

```
import java.util.Scanner;
public class Task15Way01 {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        System.out.print ("Please enter a num");
        int x= sc.nextInt();
        int count = 0;

        for (int i=1; i<=x; i++){
            if (x % i == 0) {
                count ++;
            }
        }
        if (count == 2) {
            System.out.print ("Prime number");
        }
        else {
            System.out.print ("NOT a prime number");
        }
    }
}
```

* The only even prime number is 2, All other even numbers are divisible by 2.

Task 15 Way 02

```
import java.util.Scanner;
public class Task15Way02 {
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);
        int n = sc.nextInt();
        System.out.print ("Enter a number");
        int num = sc.nextInt();
        int count = 0;
        for (int i=2 ; i<num ; i++){
            if (num % i ==0){
                count ++;
            }
        }
        if (count ==0){
            System.out.print ("Prime number");
        }
        else {
            System.out.print ("Not prime number");
        }
    }
}
```

Task 19

```
public class Task01 {
    public static void main (String [] args) {
        int [] a = new int [] {10, 30, 20, 50, 40};
        int max = a[0], min = a[0], sum = a[0];

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

            if (a[i] > max) {
                max = a[i];
            }
            else if (a[i] < min) {
                min = a[i];
            }
            sum += a[i];
        }

        int avg = sum / a.length;

        System.out.println ("Min: " + min);
        System.out.println ("Max: " + max);
        System.out.println ("Avg: " + avg);
    }
}
```

* avg/sum → better to take double instead of int.

Task20

```
public class Task20 {
    public static void main (String [] args) {
        int [] a = new int [] {10, 30, 20, 50, 40};

        int sum = a[0];
        int count = 0;

        for (int i=1 ; i<a.length ; i++) {
            sum += a[i];
        }
        int avg = sum / a.length;

        for (int j=1; j<a.length; j++) {
            if (a[j] > avg) {
                count ++;
            }
        }
        System.out.println(count + " students are better than avg");
        System.out.println ("They received the following marks");
        for (int k=1 ; k<a.length ; k++) {
            if (a[k] > avg) {
                System.out.println (a[k]);
            }
        }
    }
}
```

Task 21

```
public class Task21 {
    public static void main (String [] args){
        int [] a = new int [] { 10, 30, 20, 50, 40};
        int max = a[0], maxLoc = 0;

        for (int i=1; i<a.length; i++){
            if (a[i] > max) {
                max = a[i];
                maxLoc = i;
            }
        }
        System.out.println ("Largest number is " +max);
        System.out.println ("Largest number was found at
                            location " + maxLoc);
    }
}
```

* for location assign the variable max/min Loc *

* Swapping e • temp variable / backup variable
• max/min Loc variable

Task 22

```
public class Task22 {
    public static void main (String [] args) {
        int[] a = new int[] {10, 30, 20, 50, 40};
        int max = a[0], maxLoc = 0;
        for (int i=1 ; i<a.length; i++) {
            if (a[i] > max) {
                max = a[i];
                maxLoc = i;
            }
        }

        int backup = a[0];
        a[0] = max;
        a[maxLoc] = backup;

        for (int i=0 ; i<a.length; i++) {
            System.out.print (a[i] + ", ");
        }
    }
}
```

Task 23

```
public class Task05 {
    public static void main (String [] args){
        int [] a = new int [] { 50, 30, 20, 10, 40};
        int max = a [1], maxLoc = 1;

        for (int i=2; i < a.length ; i++) {
            if (a[i] > max){
                max= a[i];
                maxLoc = i;
            }
        }

        int temp = a [1];
        a[1] = max;
        a[maxLoc] = temp;

        for (int i=0; i < a.length; i++){
            System.out.print (a [i] + " , ");
        }
    }
}
```

Descending Order

Task 25

Way 01

Important

```
public class Task07 {
    public static void main (String [] args) {
        int [] a = new int [] { 10, 20, 30, 50, 40 };

        for (int i=0; i<a.length ; i++) {
            for (int j= i+1 ; j<a.length; j++) {
                if ( a [j] > a[i]) {
                    int temp = a [j];
                    a [j] = a [i];
                    a [i] = temp;
                }
            }
        }
        for (int i=0; i <a.length ;i++) {
            System.out.print (a [i] + ", ");
        }
    }
}
```

* just change the sign for ascending order

Task 25
Way 02

```
public class Task25 {
    public static void main (String [] args) {
        int [] a = new int [] { 10, 30, 20, 50, 40 };
        for (int x=0; x < a.length -1 ; x++) {
            int max = a [x];
            int maxLoc = x;
            for (int c= x+1 ; c < a.length; ++c){
                if ( a[c] > max) {
                    max = a[c];
                    maxLoc = c;
                }
            }
            int y = a [x];
            a[x] = max;
            a [maxLoc] = y;
        }
        for (int c=0; c<a.length; ++c) {
            System.out.print ( a[c] + ", ");
        }
    }
}
```

Task 26

```
import java.util.Scanner;
public class Task26{
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);
        System.out.print ("please enter a value for n");
        int n= sc.nextInt();

        int [] a = new int [n];
        for (int i= 0; i <a.length; i++){
            System.out.println ("please enter a number");
            a [i] = sc.nextInt();
        }
        for (int i=0; i<a.length; i++){
            for (int j= i+1 ; j <a.length; j++){
                if (a [j] > a[i]){
                    int temp= a [j];
                    a[j] = a [i];
                    a[i] = temp;
                }
            }
        }
        for (int i=0 ; i<a.length; i++){
            System.out.print (a[i] + " ");
        }

        double median;
        if (a.length % 2 == 0){
            int x = a.length /2;
            int y = x-1;
            median = (a[x] + a[y]) /2.0;
        }
```

```
else {
    int x = a.length /2;
    median = a[x];
}
System.out.println ("Median: " +median);
}
}
```

Array → address → no. of boxes

index of an array starts from zero

Example:

```
int[] a = new int[5];
int x = sc.nextInt();
for (int index = 0; index < 5; index ++) {
    arr [index] = sc.nextInt();
}
```

Arrays are the types of variables that lets us store and access several values under the same name.

```
int[] a = new int[n];
for (int i = 0; i < n ; i++) {
 a [i] = sc.nextInt();
 }
```

Input loop

```
for (int j = 0; j < a.length; j++) {
        System.out.println(a[j]);
}
```

Printing loop

Reverse order printing loop

```
for (int i = a.length - 1 ; i >= 0 ; i--) {
        System.out.println(a[i]);
}
```

Q. Take 10 numbers from the user and print their sum. Then print the number in sequence / in given serial.

Ans:

```
import java.util.Scanner;
public class Task {
    public ~~class~~ static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        int sum = 0;
        int [] a = new int [10];
        for (int i=0 ; i < a.length ; i++) {
            a [i] = sc.nextInt ();
            sum = sum + a [i] ;
        }
        System.out.print ("sum: " + sum);

        for (int j=0; j < a.length; j++) {
            System.out.print ( a [j] + ",");
        }
    }
}
```

	j
	0
	1
	2
	3
	4
	:
	:

i = index

a [i] = value of index

Q. Take 10 numbers. Add all the even numbers. Print average of the even numbers.

```
int [ ] a = new int [10];
int sum = 0, evenCount = 0;
for (int i=0; i< a.length; i++){
    a[i] = sc.nextInt();
    if (a[i] % 2 == 0){
        sum = sum + a[i];
        evenCount++;
    }
}

double average = sum / evenCount;
System.out.println("Avg is " + average);
```

Lab 11

Task 1

```
import java.util.Scanner;
public class Task01{
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);

        int[] a = new int [5];
        for (int i=0 ; i<a.length ; i++){
            System.out.print ("Enter a number");
            a[i] = sc.nextInt();
        }

        int max = a[0];
        for (int j=1 ; j <a.length; j++){
            if (a[j] > max){
                max = a[j];
            }
        }

        System.out.print ("Largest number is "+max);
    }
}
```

Task04 Ascending → (low to high)

Task 4

```
import java.util.Scanner;
public class Task04{
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);

        int [] a = new int [5];
        for (int i=0 ; i<a.length; i++) {
            System.out.print ("Pls enter a num");
            a[i] = sc.nextInt();
        }


        for (int i=0 ; i<a.length ; i++){
            for (int j= i+1 ; j<a.length; j++){
                if (a[j] < a[i]) {
                    int temp = a[j];
                    a[j] = a[i];
                    a[i] = temp;
                }
            }
        }

        for (int i=0; i<a.length; i++){
            System.out.print (a[i] + " ");
        }
    }
}
```

RIGHT HAND
>
peace

Task 05 Descending (high to low)

```
import java.util.Scanner;
public class Task05Descending {
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);

        int [] a = new int [5];
        for (int i=0; i<a.length ; i++){
            System.out.print ("Please enter a number");
            a[i] = sc.nextInt();
        }
        for (int i=0; i<a.length; i++){
            for (int j= i+1; j<a.length; j++){
                if ( a[j] > a[i]){
                    int temp = a[j];
                    a[j] = a[i];
                    a[i] = temp;
                }
            }
        }
        for (int i=0; i<a.length; i++){
            System.out.print (a[i] + " ");
        }
    }
}
```

Median

nested loop
to sort into an order

Task 06

```
import java.util.Scanner;
public class Task06{
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);
        System.out.print ("Please enter a value for n");
        int n = sc.nextInt();

        int [] a = new int [n];
        for (int i=0; i<a.length; i++){
            System.out.print ("please enter num");
            a[i] = sc.nextInt();
        }

        for (int i=0; i < a.length; i++){
            for (int j= i+1; j<a.length; j++){
                if (a[j] > a[i]){
                    int temp= a[j];
                    a[j] = a[i];
                    a[i] = temp;
                }
            }
        }

        double median;
        if (a.length %2 == 0){
            int x = a.length/2;
            int y = x-1;
            median = (a[x] + a[y]) /2.0;
        }
```

```
else {
        int x = a.length /2;
        median = a[x];
    }
    System.out.print("Median: " + median);
  }
}
```

k08

```
import java.util.Scanner;
public class Task08{
    public static void main (String []args){
        Scanner sc = new String []:
        { "zero", "one", "two", "three", "four", "five", "six",
          "seven", "eight", "nine"};

        System.out.print ("Pls enter a num between 0 and 9);
        int num = sc.nextInt();

        for (int i=0; i<a.length; i++){
            if (num==i){
                System.out.print (a[i]);
            }
        }
    }
}
```

Lab 10

2

```
import java.util.Scanner;
public class Task02{
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);
        int [] a = new int [10];

        for (int i=0; i<a.length; i++){
            System.out.print ("Pls enter a num");
            a[i] = sc.nextInt();
        }

        System.out.print ("Please enter a num betn 0 & 9");
        int x = sc.nextInt();

        for (int i=0; i<a.length; i++){
            if (x==i){
                System.out.print (a[i]);
            }
        }
    }
}
```

Rectangle Number

```
import java.util.Scanner;
~~pab p~~
public class RectangleNumber {
    public static void main (String [] args){
        Scanner sc = new Scanner (System.in);

        System.out.print ("Enter a number for ROW");
        int row = sc.nextInt();
        System.out.print ("Enter a number for COLUMN");
        int column = sc.nextInt();

        for (int c=0; c<row; c++){
            for (int i=1; i<=column; i++){
                System.out.print (i);
            }
            System.out.println();
        }
    }
}
```

for RECTANGLE-STAR put ("*")

Reverse Order

Task 3

```
import java.util.Scanner;
public class Task03{
    public static void main (String []args){
        Scanner sc = new Scanner (System.in);
        int []a = new int [10];
        for (int i=0 ; i<a.length ; i++){
            System.out.print ("Pls enter a num");
            a[i] = sc.nextInt();
        }
        for (int j= a.length-1 ; j>=0 ; j--){
            System.out.print (a[j] + " ");
        }
    }
}
```

Output

4 2 6

6 2 8

Practice Problems → Tracing Explanations → Array → AT v2

Trace Table

my Array										index 1	index 2	Output ~~index 3~~
0	1	2	3	4	5	6	7	8	9			
4	5	6	7	8						0	0	4
	7	7	7	7						1	1	7
			11	9 10						2	1	11
				17						3	2	17
										4	1	
										5	2	
											3	
											1	
											2	
											3	
											4	
											1	

mA[1] = 1 + 3
= 4

mA[2] = 2 + 3
= 5

m[2] = m[2] + m[1] − 2
= 5 + 4 − 2
= 7

ind2 = 1 + 1 = 2

m[3] = 3 + 3
= 6

m[3] = m[3] + m[1] − 3
= 6 + 4 − 3
= 7

= m[3] + m[2] − 3
~~= 6 + 7 − 3~~
= 7 + 7 − 3
= 11

m[4] = m[4] + m[1] − 4
= 7 + 4 − 4 = 7

m(4) = m(4) + m(2) − 4
= 7 + 7 − 4
= 10

m(4) = m(4) + m(3) − 4
= −4 + 11 − 4 = 3

Tracing practice

$m[1] = 1 + 4 = 5$

$1 < 1$ x

$m[2] = 2 + 4$
$= 6$

$m[2] = b[2] + m[1] - 2$
$= 6 + 5 - 2$
$= 9$

$m[3] = 3 + 4$
$= 7$

$m(3) = b(3) + m(1) - 3$
$= 7 + 5 - 3$
$= 9$

$m(3) = b(3) + m(2) - 3$
$= 9 + 7 + 9 - 3$
$= 13$

$= 9 + 9 - 3$
$= 15$

$m(4) = b(4) + m(3) - 4$
$= 14 + 15 - 4$
$= 25$

$m(5) = 5 + 4 = 9$

$m(5) =$

$m(5) = b(5) + m(1) - 5$
$=$

$m[4] = 4 + 4$
$= 8$

$m[4] = b(4) + m(1) - 4$
$= 8 + 5 - 4$
$= 9$

$m(4) = b(4) + m(2) - 4$
$= 9 + 9 - 4$
$= 14$

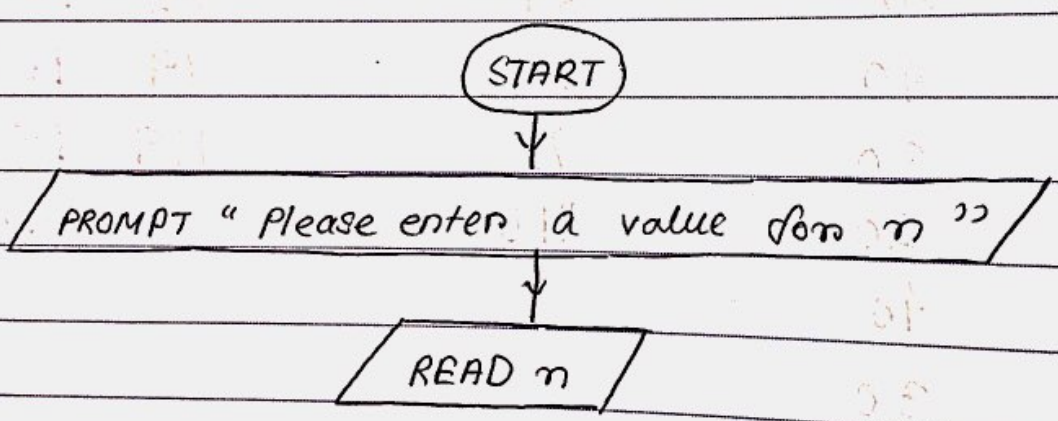
Midterm Set B

ns 2:

int x	int y	Int sum	output
1	0	0	24 ✓
3	40	23	27 ✓
5	30	26	38 ✓
7	20	35	47 53
9	40	44	59 80
	30	54	69 113
	20	64	85 138
	40	78	99 161
	30	92	119 186
	20	110	209
	40		
	30		
	20		
	40		
	30		

ns 2:

int x	int y	int sum	output
			24
1	0	0	27
	40	23	~~30~~
	30	26	
	20	~~29~~	
3	40	35	38
	30	44	47



MAIL

21. x	y	sum	output
0	0	0	23 ✓
2	-3 ✓	23 ✓	46 ✓
4	40	46 ✓	52 ✓
6	30	52 ✓	75 ✓
8	20	75 ✓	98 ✓
10	-1 ✓	98	121 ✓
	40	121	144 ✓
	30	144	167 ✓
	20	167	190 ✓
	1	190	213 ✓
	40	213	
	30		
	20		
	3		
	40		
	30		
	20		
	5		
	40		
	30		
	20		

- There are (8) primitive (simple) types of data defined by Java.

int, double, float, boolean, byte, short, long, char

Integer types	width
long	64
int	32
short	16
byte	8

(lisby)

* The smallest integer type is byte.

* The most commonly used integer type is int. It is a signed 32-bit type that has a range from −2,147,483,648 to 2,147,483,647

* There are two kinds of floating-point types, float and double, which represent single- and double-precision numbers, respectively.

Name	Width in Bits
double	64
float	32

* Double precision uses 64 bits to store a value; is actually faster and has a larger degree of precision.

Usage of double variables to compute the area of a circle.

```
class Area {
    public static void main (String [] args) {
        double pi, r, a;

        r = 10.8;   // radius of a circle
        pi = 3.1416; // pi, approximately
        a = pi * r * r; // compute area

        System.out.println (" Area of circle is " + a);
    }
}
```

$j = --k$

MAIL

~~temp~~ ~~sum~~ y

Q4.

String test	int i	j	k
" "	0	0	15
"-->"	1	14	14
0 + 14 + "-->" + 0+14	2	13	13
= "14--> 014"	3	12	
	4	11	12
	5	10	
0 + 13 + "14--> 014" + 0 + 13		10 12	11
= "1314 --> 014013"		13 11	10
		12 10	
0 + 12 + "1314--> 014013" + 0+12		11 11	
= "121314 --> 014013012"		10 10	
		10	
0 + 11 + "121314 --> 014013012" + 0 + 11			
= "11121314 --> 014013012011"			

output

14 --> 014

1314 --> 014013

121314 --> 014013012

11121314 --> 014013012011

213 --> 213

212 213 --> 213212

2112 213 --> 213212211

14 --> 113

13 -->

1314 --> 113112

121314 --> 113112111

14 --> 212

1314 --> 212211

14 --> 311

i	test
①	"--->"
	1 + 13 + "--->" + 1+13
	= 14 + "--->" + 1+13
	= "14 ---> 113"
	1 + 12 + "14 ---> 113" + 1 + 12
	= "13 14 ---> 113 112"
	1 + 11 + "1314 ---> 113112" + 1+11
	= "1213 14 ---> 113 112 111"
②	"--->"
	2 + 12 + "--->" + 2 +12
	= "14 ---> 212"
	2 +11 + "14 ---> 212" + 2 +11
	= "13 14 ---> 212 211"
③	"--->"
	3 + 11 + "--->" + 3 + 11
	= "14 ---> 311"
④	"--->"

Lab 12 Nov 30

Task 1 → Human Tester

h_1

int age	double height
21	5.5
22	

h_2 / h_1 → same address from line 13

int age	double height
21	2.5
21	5.5
22	6.5
23	
24	

Output

21
5.5
2.5
22
22
5.5
23
6.5
24
6.5

for the post-increment part,

$h_1 = 5$

$h_2 = h_1++$

tahole h_1'er value h_2 teh boshbe i.e. $h_2 = 5$ hoilo, then h_1 er value 1 kore barbe ∴ $h_1 = 6$ hoye jabeh.

So in the end, $h_2 = 5$ and $h_1 = 6$.

In short, h_2 teh h_1 er value boshbe first e and then h_1 er value change hobe.

Task 2

variables

int sum	int y
25	0
80	11

methods

method A

int x	int y
0	0
18	7

method B

int x
0
0 + 33 + 11 = 44

Output

18 7 25

44 11 80

✓

* main method'e dukbar method A run হচ্ছে hoiseh

Task 03 Puzzle

variables	methods			
int x	method A	method B (int y)	method B (int z, int x)	
5	int z	int y	int z (value of x)	int x (value of z)
10				
30	18	5	30	63
	63	20	31	64

z = x + method B (x)
= 5 + method B(x)
= 5 + 13
= 18

Or

z = method B(z+2) + x
⇒ z = method B(20) + x
= 33 + 30
= 63

z = z + 1
= 30 + 1
= 31

x = x + 1
= 63 + 1
= 64

P.T.O.

Output

10 5
10 18
30 20
30 63
31 64
30 63

10 5
10 18
30 20
30 63
31 64
30 63

Test 4

variables

int sum	int y
0	0
1	0
6	1
	2
	5
	11
	23

method A()

int x	int z
0	0
1	1
3	2
~~14~~	
12	

method B (int m, int n)

int x	int sum	int m	int n
0	0	1	0
-4	1	3	2
0	0	12	11
-2	5		
0	0		
7	23		

Output-

```
1   0   0
-4  1   1
3   2   1
-2  5   5
14  11  6
7   23  23
```

sum = sum + methodB(x,y)
= 0 + methodB(x,y)
= methodB(x,y)
= 1

sum = 1 + methodB(x,y)
= 1 + 5
= 6

sum = 6 + methodB(x,y)

1. Q. Print the following sequence of values in loops (Lab 04)

d) 18, -27, 36, -45, 54, -63

```
public class Task01d {
    public static void main (String [] args){
        int sign = 1;
        for (int count = 18; count <= 63; count += 9){
            System.out.print (count * sign + " ");
            sign *= -1;
        }
    }
}
```

Tracing Scope

int x	int y
4	1
6	2
8	7
6	9
8	10
18	17
20	18
38	37

method A()

method B()

Output

6	2	✓
8	7	✓
8	10	✓
18	17	✓
20	18	✓
38	37	✓

```
public class SingleArraySum {
    public static void main (String [] args) {
        int scores [] = { 4, 2, 3, 5 };
        int sum = 0;

        for (int i=0 ; i < scores.length ; i++)
            sum= sum + scores [i];
        }
        System.out.println (sum);
    }
}
```