

LAB-1

Date _____
Page _____

```
import java.util.Scanner;
```

```
public class QuadraticSolver {
```

```
    public static void main(String args[])
```

```
    {
```

```
        Scanner in = new Scanner(System.in);
```

```
        double a = scanner.nextDouble();
```

```
        System.out.println("Enter the coefficient  
b:");
```

```
        double b = in.nextDouble();
```

```
        System.out.println("Enter the coefficient  
c:");
```

```
        double c = in.nextDouble();
```

```
        double d = in.nextDouble();
```

```
        double discriminant = b * b - 4 * a * c;
```

```
        if (discriminant > 0)
```

```
            double root1 = (-b + Math.sqrt(discriminant))
```

```
                / (2 * a);
```

```
            double root2 = (-b - Math.sqrt(discriminant))
```

```
                / (2 * a);
```

```
            System.out.println("Real Solutions:");
```

```
            System.out.println("Root 1: " + root1);
```

```
            System.out.println("Root 2: " + root2);
```

else

{

System.out.println("No real solutions.");

}

}

}

}

OUTPUT:

3.0000000000000004

Enter coefficient = a:1.000

Enter coefficient + b:2.000

Enter coefficient + c:1.000

Real solutions:

Root 1: -1.0

Root 2: -1.0

0.0000000000000002

b

?

1.0000000000000002

2.0000000000000002

1.0000000000000002

0.0000000000000002

-1.0000000000000002

-1.0000000000000002

1.0000000000000002

2.0000000000000002

1.0000000000000002

-1.0000000000000002

-1.0000000000000002

1.0000000000000002

2.0000000000000002

1.0000000000000002

-1.0000000000000002

-1.0000000000000002

Date 9/1/23

Page

Program (2::19) : " "

```

import java.util.Scanner;
class Subject {
    int marks;
    int credits;
    float grade;
}

int subjectMarks;
int credits;
float grade;

```

class Student {

Subject subjects[2];

String name, usn;

double SGPA;

Scanner s = new Scanner (System.in);

Student() { // constructor }

subjects = new Subject[2];

for (int p = 0; p < 2; p++) {

subjects[p] = new Subject();

}

void getd() {

System.out.println ("Enter the student name");

name = s.nextLine();

System.out.println ("Enter the usn:");

usn = s.nextLine();

void getmarks() {

for (int p = 0; p < 2; p++) {

System.out.println ("Enter details of

for subject: " + (i + 1));
 System.out.println ("Enter Marks:");
 Subject[i].SubjectMarks = s.nextInt();
 System.out.println ("Enter the credit:");
 Subject[i].credits = s.nextInt();

if (Subject[i].SubjectMarks >= 90) {
 Subject[i].grade = 10;

}

else if

(Subject[i].SubjectMarks >= 80) {
 Subject[i].grade = 9;

}

..

..

if (Subject[i].SubjectMarks <= 20) {
 Subject[i].grade = 2;

} // Last of if statement

for (i = 0; i < 9; i++) {

System.out.println ("Subject " + i + " grade: " + Subject[i].grade);

} // getMarks();

void getSGPA () {

double totalCre, sum, m;

for (int i = 0; i < 9; i++) {

totalCre += Subject[i].credits;

sum += (Subject[i].grade *
 Subject[i].credits);

} // Last of for loop

SGPA = sum / totalCre;

```
void display(); { } // empty function
```

System.out.println ("In Result : \n

name: \n" + name
+ " usn: \n" + usn
+ " SC_PA: \n" + SC_PA);

9 11 class student

Class LAB2

• (c) $\text{P}(\text{left}) = \frac{1}{2}$ $\text{P}(\text{right}) = \frac{1}{2}$

public static void Main()

Student *s1 = new Student();

81. `getd();`

St. 2 get marks (1) = 10 with

31. GetSGPA();

```
std::display(G);
```

3

(ex transversal) 19

91172

• (2 to 3) N

$$\text{angle} - d - \gamma = \text{start angle}$$

:(a + b)

"1205332212 1359") with ID 3113-163099

1995-07-11 1995-07-11 1995-07-11

LAB-3

Date: 1/1
Page: 1

```
import java.util.Scanner;  
class Book {  
    String name, author;  
    float price, num_pages;  
    Scanner in = new Scanner(System.in);  
    Book(String name, String author,  
        float price, float num_pages)  
    {  
        this.name = name;  
        this.author = author;  
        this.price = price;  
        this.num_pages = num_pages;  
    }
```

```
void getd()  
{  
    System.out.println("Enter the name of  
        the book:");  
    name = in.nextLine();  
    System.out.println("Enter the author:");  
    author = in.nextLine();  
    System.out.println("Enter price:");  
    price = in.nextInt();  
    System.out.println("Enter no. of pages:");  
    num_pages = in.nextInt();  
}
```

```
void display(){
```

```
    System.out.println("name : " + author:  
        " " + price : " " + num_of_g  
        pages : " " + name  
        + author + price  
        + num_pages );
```



public String toString()

return name + author + price + num-pages;

3. Extend to our Book class

class Lib3 {

public static void main (String args[])

{

int n;

Book b[];

Scanner in = new Scanner (System.in);

System.out.println ("Enter the total no. of books");

n = in.nextInt();

b = new Book[n];

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

{

System.out.println ("Enter details");

String name = in.next();

String author = in.next();

int price = in.nextInt();

int num-pages = in.nextInt();

b[i] = new Book (name, author, price, num-pages);

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

{

System.out.println (b[i].name + b[i].author +

b[i].price + b[i].num-pages);

}

}

}

UASER

INPUT

Registration Card

OUTPUT: Registration card

Enter the total no. of books:

2

Enter details for book 1:

Merchant of venice

Shakespeare

1000

300

1000
300

(1000+300)*100 = 130000

Enter details for book 2:

If - Mr. never dies:

Sidney - sheldon

800

270

1000
270

(1000+270)*100 = 127000

Book details 1:

Merchant of venice

Shakespeare

1000

300

1000
300

(1000+300)*100 = 130000

Book details 2:

If - Mr. never dies:

Sidney - sheldon

800

270

1000
270

(1000+270)*100 = 127000

LAB - 4:

Date _____
Page _____

```
import java.lang.Math;  
import java.util.Scanner;  
abstract class Shape {  
    protected int pnt1, pnt2;  
    abstract void printArea();  
    Scanner in = new Scanner(System.in);  
}
```

class Rectangle extends Shape {

```
    Rectangle() {  
        System.out.println("Rectangle object  
        created");  
        void printArea() {  
            int area;  
            System.out.println("Enter the length  
            and Breadth of the rectangle:");  
            pnt1 = in.nextInt();  
            pnt2 = in.nextInt();  
            area = pnt1 * pnt2;  
            System.out.println("Area is " + area);  
        }  
    }
```

class Triangle extends Shape {

```
    Triangle() {  
        System.out.println("Triangle object  
        created");  
    }
```

```
    void printArea() {  
        double Area;  
        System.out.println("Enter the base  
        and height:");  
    }
```

```
    rnt1 = in.nextInt(); // input for first  
    rnt2 = in.nextInt(); // input for second  
    area = 0.5 * rnt1 * rnt2; // calculate  
    System.out.println("Area of  
    the triangle is: " + area);  
    // Output: 3 4  
    // Area is 6.0
```

2. Create another program with class Lab4

```
class Lab4 {  
    public static void main (String args []) {
```

```
        Rectangle obj1 = new Rectangle();  
        obj1.printArea();
```

```
        Triangle obj2 = new Triangle();  
        obj2.printArea();
```

Output with length 3 and breadth 4

1. Rectangle object created
2. Triangle object created

OUTPUT:

1. Rectangle object created

2. Triangle object created

Enter the length and breadth:

3 4

Area is 12.0

Triangle object created

Enter the base and height:

4 5

Area of the triangle: 10.0

3. Create another program with class Lab4

and add following code

1. Create a class Lab4

```
import java.util.Scanner; public  
class Account {
```

{

```
String customerName; double
```

```
long accountNumber;
```

```
String accountType;
```

```
double balance;
```

```
Account (String customerName,
```

```
long accountNumber,
```

```
String accountType,
```

```
double balance)
```

{

```
this.name = customerName;
```

```
this.accnum = accountNumber;
```

```
this.acctype = accountType;
```

```
this.balance = balance;
```

{}

```
void deposit (double amount)
```

```
balance += amount;
```

```
System.out.println ("Deposit successful");
```

```
String;
```

~~```
void display ()
```~~~~```
{
```~~

```
System.out.println (accountNumber +
```

```
accountType +
```

```
customerName +
```

```
balance);
```

{}

```
class CurAcc extends Account
```

{}

double minimum bal = 1000; trans
double service = 100

public CurrentAccount (String cname,
String ctype,
long accnum,
double balance)

{ withdraw (double) throws
Insufficient funds

super (customerName, accountNumber,
accountType = "Current Account",
balance);

balance = 3000.25

balance = 2000.25

class Savings extends Account
(balance = 5000.00)

double interestRate = 0.05;

public class Savings { String cname,
String ctype,
long accnum,
double balance)

super (customerName, accountNumber,
"Savings Account", balance);

+ withdraw (double) throws Insufficient

+ deposit (double)

public class Lab5 {

{ public static void main (String args)
{

Scanner fin = new Scanner (System.in);

CurAcc obj1 = new CurAcc ("Ankit",
1234567, 1500);

obj1. display();
obj1. withdraw();

System.out.println ("Enter the deposit
amt: ");

double dep = in. nextDouble();
obj1. deposit (dep);

Savings obj2 = new Savings ("SS",
45678910, 500.0);
obj2. display();

OUTPUT:

Account Number: 12345678910

Customer Name: Ankit

Account type: Current Account

Current balance: 1500.0

Enter the deposit amount:

100

Deposit is successful

Account number: 45678910

Customer name: SS

Account type: Savings Account

Current Balance: 500.0

LAB-6

generics:

class Stack (A, \cup)

A (a, b, c, d, e)

v f,g,h,q,g;

void SetA (A = a, A = b, A = c, A = d, A = e) {
this - a = a;

$$\text{rhs} - a = a$$

$$\text{this} \cdot b = b';$$

thes + cb = ca; 191?

thes. d = d (593)

$$\text{this} \cdot e = e \quad ; \quad (9) 9$$

9

$$100 \text{ kg} = 100000 \text{ g}$$

10) $b_2 \rightarrow p = 100$

void Setup(uf, ug, uh, uq, ug)

longer was his time spent writing

26.2.2003 10:31:03

274 this of = f j 283 27169

$$20 + 7 \text{ has } 9 = 9$$

Thus, $n = 17$ is a possible value.

Explain this. $Q = Q$

3. ~~this~~ $g = q$ is a monoid

~~CLT Play~~ 6/1/07

Class Generics

```
public static void main (String  
args []) {
```

```
Stack<Integer, Double> stack = new Stack<  
    Integer, Double>();
```

`<Integer, Double> ()`

Obj 1. ~~set A(1, 1, 1, 1, 1)~~;

obj1.setv(1.68, 1.69, 1.70, 1.71, 1.72);

System.out.println(obj1.a);

System.out.println(obj1.b); (P)

System.out.println(obj1.c);

System.out.println(obj1.f);

System.out.println(obj1.g);

System.out.println(obj1.h); (Q)

9)

Java

Method 1 (a)

STRINGS

Method 2

OUTPUT

Method 3

1) Java

V

1. (a)

path

File

Java

2

Method 1

ABCDEF

3

Method 2

CDEF

4

Method 3

2) 3

abc abc

5

Method 1

Method 2

Method 3

3) Dimensions are 10.0 by 14.0 by 12.0

Box b: Dimensions are 10.0 by 14.0 by 12.0

4) Bms

b) Bms equals Bms → true

Bms equals cdg → false

Bms equals BMS → true

Bms equals BMS → false

5) 72

H

89) substrings are matched

899) It is true

It is true

10) Hello.equals Hello \rightarrow True

Hello == Hello \rightarrow False

11) batch

van

apple

lou xmas

ball

TUB batch

cat

ree

dog

ent

1

v

tree

2

same

gun

3

730394

hen

4

791

pcu

5

gvg

6

8

Refe

7

releas

lft

8

mon

9

785390

net

10

9-81

orange

parrot

queen

reng

star

tree

umbrella

2m3 (H)

57 (H)

14 (H)

13) This was a test. This was too. This was a test. This was too.

14) Hello world

15) Original: I go to college
Replaced: I go to commege

16) Hello world

17) Enter detail for student 1:

Reg NO: 50

Full Name: Anket.

Semester: 3

CGPA: 9.8

Similarly we enter details for 5 students

18) ~~buffer before = Hello~~

~~char At(1) before = e~~

~~buffer after = HP~~

~~char At(1) after = ?~~

H

e

e

HP 42

HPLeke 42

24 ekplogh

19) Eagle:
Eagle flies high on the sky
Eagle screeches loudly

Hawk:
Hawk soars gracefully through
the air
Hawk emits a piercing cry.

generics OUTPUT:

1. 6799

1. 636

1. 3565

1. 345

1. 879

Internal class:

```

package CIE;
import java.util.Scanner;
public class Internal extends Student {
    int p; new
    int marks[] = p[50];
    Scanner qn = new Scanner (System.in);
    public Internal () {
        System.out.println ("Internal obj");
    }
    public void getId () {
        for (p=0; p<5; p++) {
            System.out.print ("Enter
            the marks for Subject " + (p+1));
            marks[p] = qn.nextInt();
        }
    }
}

```

~~package CIE;~~

~~import CIE java.util.Scanner;~~

~~public class Student {~~

~~protected String usn, name;~~

~~protected int sem;~~

~~Scanner qn = new Scanner (System.in);~~

~~public Student () {~~

~~System.out.println ("Student
created ");~~

~~}~~

~~public void getId () {~~

~~System.out.print (" Enter USN: ");~~

~~System.out.print (" Enter Name: ");~~

~~System.out.print (" Enter Sem: ");~~

```
    usn = pn.nextIPne();
    name = pn.nextLPne();
    sum = pn.nextTnTn();
}

public void display() {
    System.out.println("Student
Details" + "Name: " + name);
    System.out.println("USN: " + usn);
    System.out.println("Semester In" + sum);
}

package SEE;
import CIE.Student;
import java.util.Scanner;
public class External extends Student {
    int[] marksSEE = new int[10];
    Scanner in = new Scanner(System.in);
    External() {
        System.out.println("External
object is created \n");
    }
    void getd() {
        for (int i = 0; i < 5; i++) {
            System.out.println("Enter
the marks for subject " + i + 1);
            marksSEE[i] = in.nextInt();
        }
    }
}
```

package Result;

import CIE.Internals;

import SEE.Externals;

public class FinalMarks {

public static void main (String args[])

int n = 2;

Internal p = new Internal[n];

External e = new External[n];

for (int p = 0; p < 1; p++) {

p[p] = new Internal();

p[p].getid();

p[p] = getid();

e[p] = new External();

e[p] = getid();

e[p] = getid();

public static void calc (Internal p,
External e) {

int ffinal = new int[5];

for (int p = 0; p < 5; p++) {

ffinalM[p] = p.pmarks[p] +

e.externalMarks[p];

System.out.println ("Course " + (p+1) +
" : " + ffinalM[p]);

LAB - 8

LAB-8
import javax.imageio.ImageScanner;

class WrongAgeException extends
Exception { }

Wrong Age Exception (Strong m)

super (m);

3

class Father {

"2" "Print page" won't do what I want

Father (Int. face) throws wrong Age
Exception: 5 1923b. 8960

qf (frage $\angle = 0$) \}

show new wrong Age Encryption

(" age (= 0) ;

else {

this page = page; about which

System.out.println("Father's Age
" + fatherAge);

tips: the face) just another

3) ~~Wiederholung~~ Wiederholung Wiederholung

class Some extends Father { } ?

Qnt. Saige (0.05%) ; 300000 ton/yr.

Son ^{int case} throws wrong Age Exception {

7:33 Super (page); 39(1) (1)(a)

9f (stage >= fage)

1. *guitar* 2. *guitar* 3. *guitar*

~~throw new WrongAgeException()~~

~~throw new WrongAgeException();~~

Son's age can't be \geq Father's age");

else { open reading with intput

{

 this.sage = sage; // read integer

 System.out.println("Son's age is: " + sage);

}

}

}

public class Lab 8 {

 public static void main (String args[])

{

 Scanner in = new Scanner (System.in);

 try {

 System.out.println ("Enter father's
 age: ");

 int nf = in.nextInt();

 Father f1 = new Father (nf);

 System.out.println ("Enter son's age: ");

 int ns = in.nextInt();

 Son s1 = new Son (ns);

 }

 catch (WrongAgeException e) {

 System.out.println ("check age: " + e);

 }

}

OUTPUT: 1. 40 is the father's age.

Enter the Father's age: 40

Father's age is: 40

2. Enter the son's age:

20

Father's age is: 40

Son's age is: 20.

~~3. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~4. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~5. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~6. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~7. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~8. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~9. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~10. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~11. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~12. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~13. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~14. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~15. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~16. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~17. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~18. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~19. Father's age is 40 and son's age is 20. Their sum is 60.~~

~~20. Father's age is 40 and son's age is 20. Their sum is 60.~~

```
public class Lab 9 {  
    static class BMSThread extends Thread {  
        public void run() {  
            try {  
                while (true) {  
                    System.out.println("BMS  
College of engineering");  
                    Thread.sleep(10000);  
                }  
            } catch (Exception e) {  
                System.out.println("Exception  
handled" + e);  
            }  
        }  
    }  
}
```

static class CSEThread extends Thread {

```
public void run() {  
    try {  
        while (true) {  
            System.out.println("CSE");  
            Thread.sleep(2000);  
        }  
    } catch (Exception e) {  
        System.out.println("Exception  
handled" + e);  
    }  
}
```

Scanned with CamScanner

```
public static void main (String  
args[]) {
```

BMSThread bms = new BMSThread();
bms.start();

```
brs! Start()
CSEThread (cse = new CSEThread(),
cse.start());
```

3) paramorfosis (paramorfosis)

9:30 a.m. - Early morning

OUTPUT: <http://www.visionmedia.com>

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE-3193: Learning with others

BMS college of Engineering
ESF

658

1998-9999 Long

—
—
—

1. *Alouatta palliata* (Linnaeus, 1758) (p. 100)

3 (cont'd) (cont'd) 3

~~other~~) adding the name?

Geometric Properties of the Ellipse

LAB-9 AWT

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
class Swing Demo {  
    Swing Demo() {  
        JFrame frm = new JFrame("Divider App");  
        frm.setSize(225, 150);  
        frm.setLayout(new FlowLayout());  
        frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        JLabel glab = new JLabel("Enter the  
        divisor and dividend");  
        JTextField atextField = new JTextField(8);  
        JTextField btextField = new JTextField(8);  
        JButton button = new JButton("calculate");  
        JLabel err = new JLabel();  
        JLabel alab = new JLabel();  
        JLabel blab = new JLabel();  
        JLabel ansLab = new JLabel();  
        frm.add(err);  
        frm.add(glab);  
        frm.add(atextField);  
        frm.add(btextField);  
        frm.add(button);  
        frm.add(alab);  
        frm.add(blab);  
        frm.add(ansLab);  
    }  
}
```

button.add ActionListener (new ActionListener()) {
 public void actionPerformed (ActionEvent evt) {

try {

int a = Integer.parseInt (aftf.getText());
int b = Integer.parseInt (bftf.getText());
int ans = a/b;

err.setText ("");

alab.setText ("A = " + a);

blab.setText ("B = " + b);

anslab.setText ("Ans = " + ans);

}

catch (NumberFormatException e) {

alab.setText ("");

blab.setText ("");

anslab.setText ("");

err.setText ("B should be non-zero!");

}

}

frm.setVisible (true);

public static void main (String args[]) {
 SwingUtilities.invokeLater (new Runnable() {
 public void run() {

new SwingDemo();

});

}

OUTPUT:

Devder App - □ X

Enter the devder
and devdend:

| | |
|-----|---|
| 500 | 6 |
|-----|---|

Calculate A = 500 B = 6
Ans = 83

2019/12/24

Functions used:

→ JFrame :

It contains the main window of the application

→ SetSize (Pnt width, Pnt height) :

It is used

to set the size of the GUI.

→ Set Layout:

Used to set the container.

→ Jlabel :

Displays text or images

→ Add frame:

Used to add a new frame.

```
class C {
    int n;
    boolean valueSet = false;
    synchronized int get() {
        while (!valueSet) {
            try {
                System.out.println("Consumer waiting \n");
                wait();
            } catch (InterruptedException e) {
                System.out.println("Exception handled");
            }
            System.out.println("got: " + n);
            valueSet = false;
        }
        System.out.println("Estimate Producer \n");
        notify();
        return n;
    }
}
```

```
synchronized void put (int n) {
    while (valueSet)
        try {
            System.out.println("Producer waiting \n");
            wait();
        } catch (InterruptedException e) {
            System.out.println("Exception handled");
        }
    this.n = n;
}
```

```
valueSet = true;  
System.out.println("Put: " + n);  
System.out.println("Infinite consumer  
(" + n + "));  
notify();
```

class Producer implements Runnable {

Producer(BlockingQueue<String> q) {

if (q.size() == 0) {
new Thread(this, "consumer").start();

if (q.size() > 0) q.add("A");

public void run() {

int q = 0;

while (q < 5) {
q = q + q.get();
System.out.println("Consumed " + q);

class Lab04 {

public static void main (String
args[]) {

8

$\delta Q = \text{new } h \& y$; ~~and~~ g

new Producer (g); ~~and~~ h

new Consumer (g); ~~and~~ h

System.out.println ("Press Control
-L to stop.");

g

: (new and old) g

out.println ("A message to the user");

Put: 0 ~~alternatively~~ h

Intimate Customer ~~h~~ g

Produce working

Get: 0 ~~start~~ h ~~working~~ h ~~working~~

Intimate producer ~~h~~ g

Put: 1 ~~alternatively~~ h

Intimate customer

Producer working

consumed: 0 ~~start~~ h ~~working~~

Cpt: 1 ~~alternatively~~ h

Intimate producer

consumed: 1 ~~start~~ h ~~working~~

~~if (a & b) and b & c then a & c~~

~~if (a & b) and b & c = a & c~~

~~if (a & b) & c = a & (b & c)~~

~~if (a & b) & c = a & c~~

~~if (a & b) & c = a & c~~

~~if (a & b) & c = a & c~~

~~if (a & b) & c = a & c~~

~~if (a & b) & c = a & c~~

class A {

synchronized void foo(B b) {

String name = Thread.currentThread().

getName();

System.out.println(name + " entered A");

try {

Thread.sleep(1000);

} catch (Exception e) {

System.out.println("A interrupted

thread");

System.out.println(name + " trying to

call: B.last()");

B.last();

synchronized void last() {

System.out.println("Inside A.last");

}

synchronized void bar(A a) {

String name = Thread.currentThread().

getName();

System.out.println(name + " entered B.bar");

try {

Thread.sleep(1000);

} catch (Exception e) {

System.out.println("B interrupted");

System.out.println("Inside B.last");

```
class Lab10b implements Runnable {  
    A a = new A(); B b = new B();  
  
    Lab10b() {  
        Thread.currentThread().setName  
            ("Main Thread");  
        Thread t = new Thread(this, "Racing  
            Thread");  
        t.start();  
        a.foo(b);  
        System.out.println("Back on main T");  
    }  
  
    public void run() {  
        b.bar(a);  
        System.out.println("Back on other T");  
    }  
  
    public static void main(String args[]) {  
        new Deadlock();  
    }  
}
```

OUTPUT:

Racing Thread entered B.bar
Main Thread entered A.foo
Main Thread trying to call B.last()
Racing thread trying to call A.last()

✓
13.2124