

Not yet
submitted
Marked out of
100
For question

Ce valoare întreagă nu se poate să fie marcată cu //? în următoarea programul de mai jos:

```
class Pair {  
    private int x;  
    private int y;  
    public Pair(int x, int y) { x = x; y = y; }  
    public void setX(int x, int y) { x = x; y = y; }  
    public void setY(int x, int y, int z) { y = y; z = z - y; }  
    public int getX() { return x; }  
    public int getY() { return y; }  
}  
  
class Main {  
    public static Pair callPair(Pair p, Pair q) {  
        q.setX(100, q.getY());  
        q.setY(10, 10);  
        int r[] = {10, 10};  
        return new Pair(r[0], r[1]);  
    }  
  
    public static void main(String[] args) {  
        int r[] = {10, 10};  
        int t[] = {10, 10};  
        Pair a = new Pair(r[0], r[1]);  
        Pair b = new Pair(t[0], t[1]);  
        Pair c = b;  
        b = callPair(a, b);  
        System.out.println(c.getX()); //?  
    }  
}
```

34

re
nit
ncă
in 1.00
are cu

Ce valoare întreagă se va tipări la linia marcată cu /* în urma execuției programului de mai jos?

```
class Ex1 extends Exception {}  
class Ex2 extends Ex1 {}  
class Main {  
    public static void m(int x) throws Ex1 {  
        if(x == 1) throw new Ex2();  
    }  
    public static void main(String argv[]) {  
        int k = 0;  
        int a = 1;  
        while (k < 2) {  
            try {  
                k++;  
                m(k);  
                a++;  
            } catch(Ex2 e) {  
                a += 2;  
            } catch(Ex1 e) {  
                a += 3;  
            } finally {  
                a++;  
            }  
        }  
        System.out.println(a);/*  
    }  
}
```

Răspuns:

7

Not yet
answered

Marked out of
3.00

T Flag question

De valoare întreagă se va tipări la linia marcată cu /* în următoarea programă.

```
class Ex extends Exception {}  
class Main {  
    public static void main(String args[]) throws Ex {  
        int i = 0;  
        int z = 0;  
        while (i < 2) {  
            try {  
                int k = 0;  
                while (k < 3) {  
                    k++;  
                    z = z + 2;  
                    if (i == 0) throw new Ex();  
                }  
                z = z + 2;  
            } catch (Ex e) {  
                System.out.println(e);  
            } finally {  
                z = z + 1;  
            }  
            i++;  
        }  
        System.out.println(z); /*
```

12

Anonim



Ce valoare întreagă se va tipări la linia marcată cu /* în urma execuției programului de mai jos?

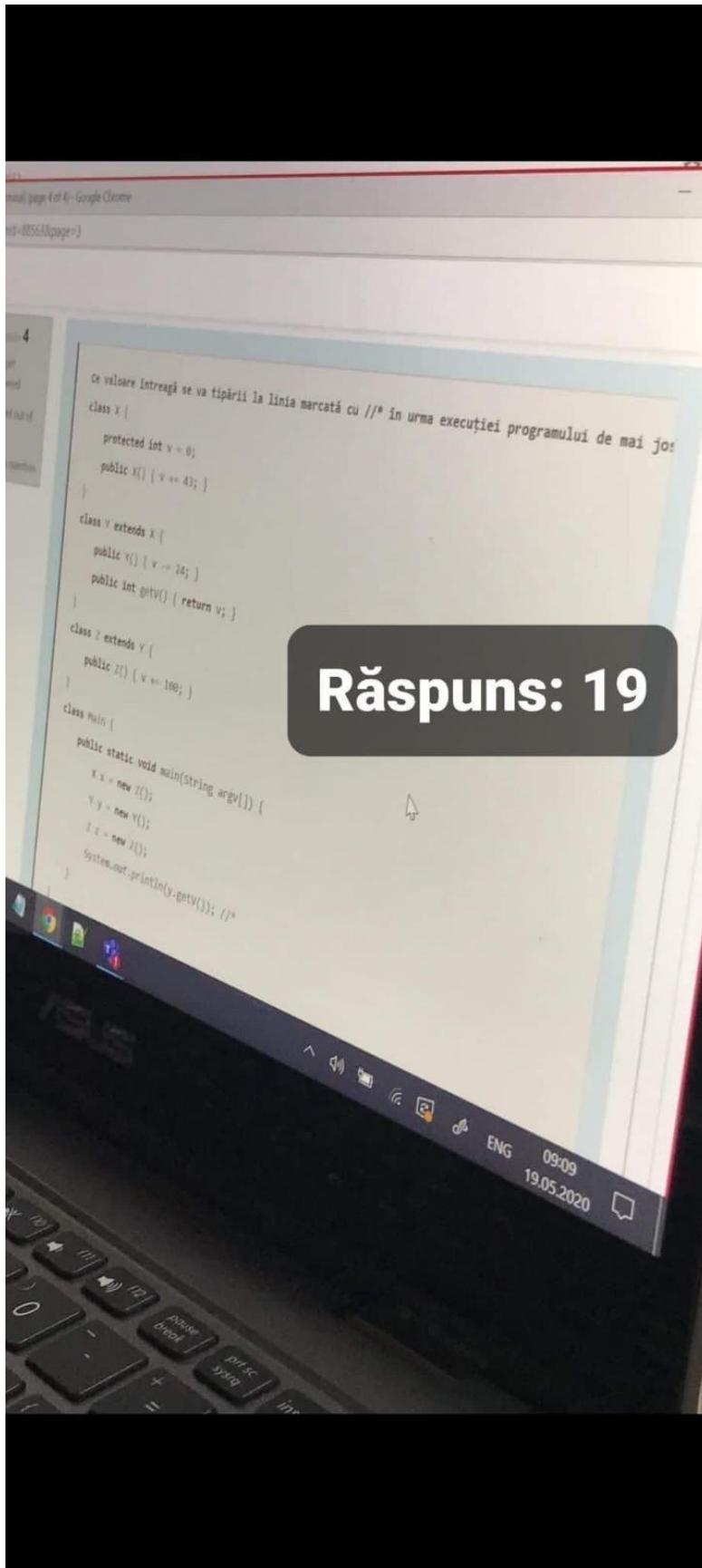
```
class A {  
    public int m(A p) { return 12; }  
    public int n(A p) { return 38; }  
}  
class B extends A {  
    public int m(B p) { return 3;}  
    public int n(A p) { return 7; }  
}  
class C extends A {  
    public int m(A p) { return 38; }  
    public int n(A p) { return 4; }  
}  
class D extends A {  
    public int m(A p) { return 1; }  
    public int n(D p) { return 2; }  
}  
class Main {  
    public static void main(String argv[]) {  
        A x = new C();  
        A y = new D();  
        D z = new D();  
        System.out.println(x.m(z) + y.n(z));/*  
    }  
}
```

R: 76



Răspuns: 19

```
ce valoare întreagă se va tipări la linia marcată cu //* în urma execuției programului de mai jos  
class X {  
    protected int v = 0;  
    public X() { v += 43; }  
}  
class Y extends X {  
    public Y() { v -= 14; }  
    public int getV() { return v; }  
}  
class Z extends Y {  
    public Z() { v += 109; }  
}  
class Main {  
    public static void main(String args[]) {  
        X x = new Z();  
        Y y = new Y();  
        Z z = new Z();  
        System.out.println(y.getV()); /*
```



12:25

4G

← W Grile Test teoretice.docx
Select one:

- a. numarul de arce pe care acesta le contine
- b. numarul de noduri pe care acesta le contine
- c. egal cu gradul grafului
- d. numarul maxim de arce incidente unui anumit nod din graf

[Clear my choice](#)

Care din urmatoarele afirmații despre codurile Huffman este adevarata:

Select one or more:

- a. Algoritmii de tip Huffman static au dezavantajul că necesită cunoașterea prealabilă a frecvențelor de apariție pentru fiecare simbol din sursă.
- b. Codarea Huffman poate prezenta pierderi (de informație), în anumite cazuri.
- c. Codarea Huffman este folosită pentru compresie.
- d. În codarea Huffman, niciun cod nu este prefixul altui cod.

Care este numărul de arce dintr-un graf complet cu N noduri?

Select one:

- a. N^2N
- b. $N-1$
- c. $N^2N/2$
- d. $(N*(N-1))/2$

[Clear my choice](#)

Pagina 7 / 20

AA

drive.google.com



12:23

4G



201221550_53 ... 883032_n.jpg



19986

Teachers:

Tutor: CRETU Vladimir - vcard

Tutor: Gheorghe Razvan - vcard

+6

More

Course start date:

15.02.2021

Enrolled users

Calendar

Grades

n.jpg



6,7

Cum se face deplasarea intr-un arbore Huffman?

- a. 0 = nu se face deplasare; 1 = in adancime
- b. 0 = stanga; 1 = dreapta
- c. 1 = stanga; 2= dreapta
- d. 0= dreapta; 1= stanga

Jump to...



19999

n.jpg



You are here

GET IT ON

Download on the



200034245 ... 275_n.jpg

200060353 ... 789_n.jpg



AA

drive.google.com



re
nit
ncă
in 1,00
are cu

Ce valoare întreagă se va tipări la linia marcată cu /* în urma executiei programului de mai jos?

```
class Ex1 extends Exception {}  
class Ex2 extends Ex1 {}  
class Main {  
    public static void m(int x) throws Ex1 {  
        if(x == 1) throw new Ex2();  
    }  
    public static void main(String argv[]) {  
        int k = 0;  
        int a = 1;  
        while (k < 2) {  
            try {  
                k++;  
                m(k);  
                a++;  
            } catch(Ex2 e) {  
                a += 2;  
            } catch(Ex1 e) {  
                a += 3;  
            } finally {  
                a++;  
            }  
        }  
        System.out.println(a);/*  
    }  
}
```

Răspuns:

7

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de compilare când sunt inserate la poziția indicată cu /*AICI*/?

```
class A {  
    public A m(int a) {  
        return new A();  
    }  
}  
class B extends A {  
    public void n(A a) {  
        return;  
    }  
/*AICI*/  
}
```

Selectați unul sau mai multe:

- ```
public int n(A a) {
 return 0;
}
```
- ```
public int n(B a) {  
    return 0;  
}
```
- ```
public A m(B a) {
 return new B();
}
```
- ```
public A m(int a) throws Exception {  
    throw new Exception("Message");  
}
```
- ```
public A m(int p) {
 return new B();
}
```

ebare  
primit  
ns încă  
at din 1,00  
trebare cu

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programului de mai jos?

```
class A {
 private int x;
 public A(int x) { this.x = x; }
 public void setX(int x) { this.x = x; }
 public int getX() { return x; }
}
class B {
 private A y;
 public B(A y) { this.y = y; }
 public void setY(A y) { this.y = y; }
 public A getY() { return y; }
}
class Main {
 public static void m(B z, B t) {
 A w = new A(4);
 z.setY(w);
 w.setX(7);
 t = new B(new A(3));
 }
 public static void main(String[] args) {
 B a = new B(new A(35));
 B b = new B(new A(33));
 m(a,b);
 System.out.println(a.getY().getX() + b.getY().getX());/*
 }
}
```

Răspuns: **42**

9 întrebare  
Nu a primit  
răspuns încă  
Marcat din 1,00  
F Intrebare cu  
flag

Considerind urmatorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de co

```
class A {
 protected int a;
 public A(int a) {
 this.a = a;
 }
}
class B extends A {
 private int b;
 /*AICI*/
}
```

Selectați unul sau mai multe:

```
public B(int a, int b) {
 super(a);
 this.b = b;
}
```

```
public B(int a, int b) {
 this.a = a;
 this.b = b;
}
```

```
public void m(int a, int b) {
 super(a);
 this.b = b;
}
```

```
public B(int b) {
 super(0);
 this.b = b;
}
```

```
public B(int a, int b) {
 this.b = b;
 super(a);
}
```



Timp ră

Considerând următorul cod Java, care din expresiile enumerate mai jos vor avea valoarea de adevar FALSE când sunt evaluate la poziția indicată cu /\*AICI\*/?

```
class A {}
class B extends A {}
class Main {
 public static void main(String argv[]) {
 A a = new B();
 B b = new B();
 Object o = new A();
 System.out.println(/*AICI*/);
 }
}
```

R: a,c

Selectați unul sau mai multe:

- A.class.equals(a.getClass())
- A.class.isInstance(b)
- o.getClass().equals(Object.class)
- a.getClass().equals(b.getClass())
- B.class.isInstance(a)

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma execuției programului de mai jos?

```
0
u
class A {
 public int m(A p) { return 8; }
 public int n(A p) { return 7; }
}
class B extends A {
 public int m(B p) { return 2; }
 public int n(A p) { return 44; }
}
class C extends A {
 public int m(A p) { return 4; }
 public int n(A p) { return 4; }
}
class D extends A {
 public int m(A p) { return 11; }
 public int n(D p) { return 30; }
}
class Main {
 public static void main(String argv[]) {
 A x = new C();
 A y = new D();
 D z = new D();
 System.out.println(x.n(z) + y.m(z));/*
 }
}
```

Răspuns: **15**

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de compilare când sunt inserate la poziția indicată cu /\*AICI\*/?

```
interface I {
 public void g();
}
interface J extends I {
 public void h();
}
abstract class A implements I {
 public void m() {}
}
class B extends A implements J {
 public void p() {}
 public void g() {}
 public void h() {}
}
class Main {
 public static void main(String argv[]) {
 J j = new B();
 A a = new B();
 /*AICI*/
 }
}
```

**R: b, d, e**

Selectați unul sau mai multe:

j.g();

a.p();

a.g();

j.m();

a.h();

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de compilare când sunt inserate la poziția indicată cu /\*AICI\*/?

```
interface I {}
interface J {}
interface IJ extends I,J {}
class A implements I {}
class B extends A implements J {}
class C extends A {}
class D implements IJ {}
class Main {
 public void doSomething(I i, J j, IJ ij) {
 A a = new A();
 B b = new B();
 C c = new C();
 D d = new D();
 /*AICI*/
 }
}
```

**R: a,b,d**

Selectați unul sau mai multe:

`j = c;`

`b = d;`

`i = c;`

`ij = jj;`

`i = ij;`

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de compilare când sunt inserate la poziția indicată cu /

Timp rămas 0:21:08

```
//A.java
package p1;
public class A {
 private void m() {}
 void n() {}
 protected void p() {}
 public void q() {}
}
```

```
//B.java
package p2;
import p1.A;
public class B extends A {
 private void r() {}
 void s() {}
 protected void t() {}
 public void k() {}
}
```

```
//C.java
package p2;
import p1.A;
public class C {
 private void z() {}
 public void v(A a, B b, C c) {
 /*AICI*/
 }
}
```

```
public void v(A a, B b, C c) {
 /*AICI*/
}
}
```

Selectați unul sau mai multe:

b.s();

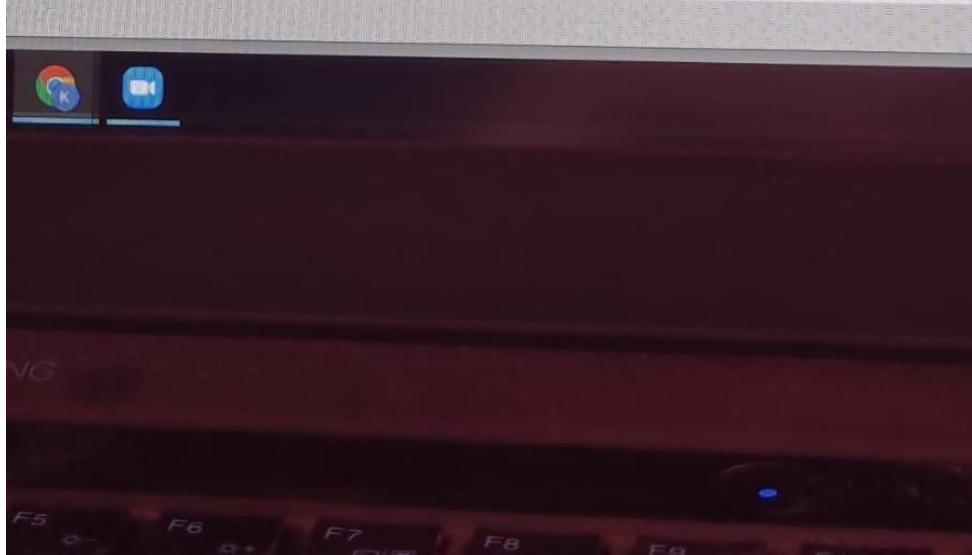
c.z();

a.p();

a.n();

b.t();

Pagina precedentă



attempt=544302&cmid=302378&page=2

You are screen sharing.

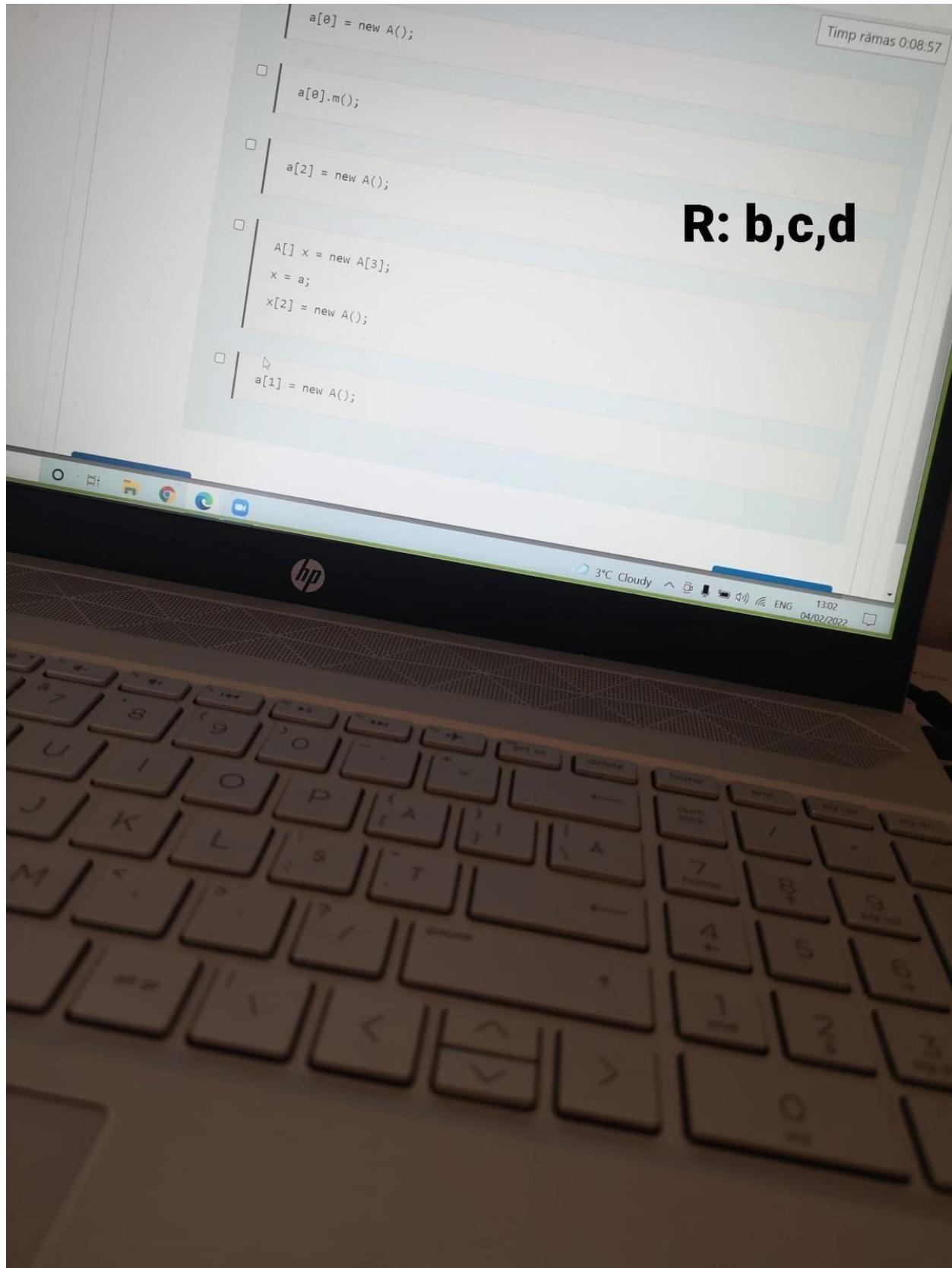
Participants Chat New Share Pause Share Annotate Remote Control More Stop Share

Time rămas 0:02:04

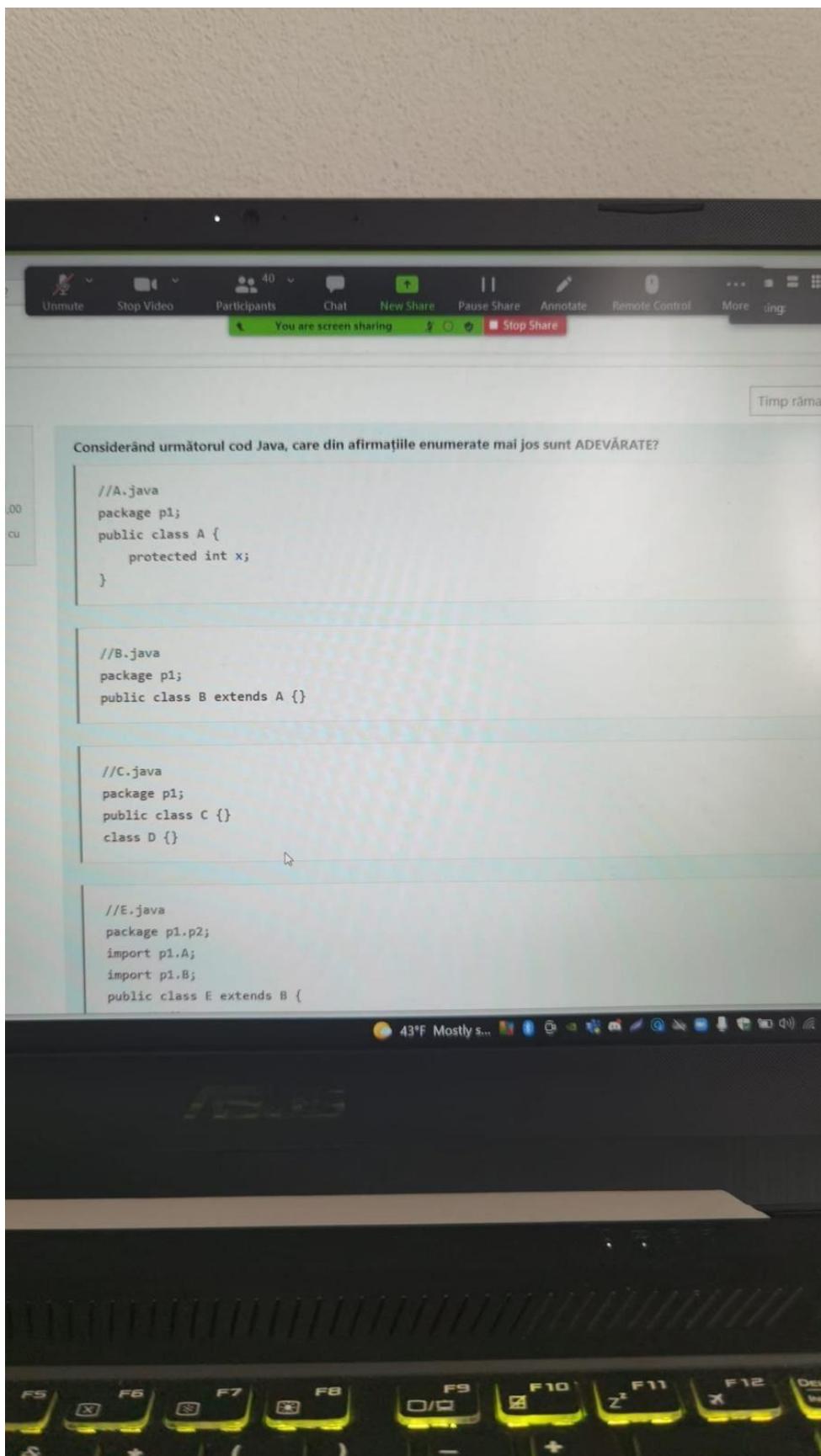
```
//E.java
package p1.p2;
import p1.A;
import p1.B;
public class E extends B {
 E() {}
 public void m(A a) {
 //1
 }
 public void n(C c) {} //2
}
```

Selectați unul sau mai multe:

- Clasa **E** poate fi instantiată în orice pachet
- Linia marcată cu //2 conține o eroare de compilare
- La linia marcată cu //1 putem accesa câmpul **X** al obiectului referit de **a**
- Clasele **A** și **B** pot fi implementate în aceeași unitate de compilare, fără a fi necesare alte modificări a programului
- Putem declara referințe de tip **D** doar în cadrul pachetului **p1**



**R: b,c,d**



7 întrebare

Nu a primit  
răspuns încă  
Marcat din 1,00  
P Întrebare cu  
flag

Timp rămas 0:09:03

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor conduce la o eroare de EXECUȚIE a programului când apar la poziția indicată cu /\*AICI\*/?

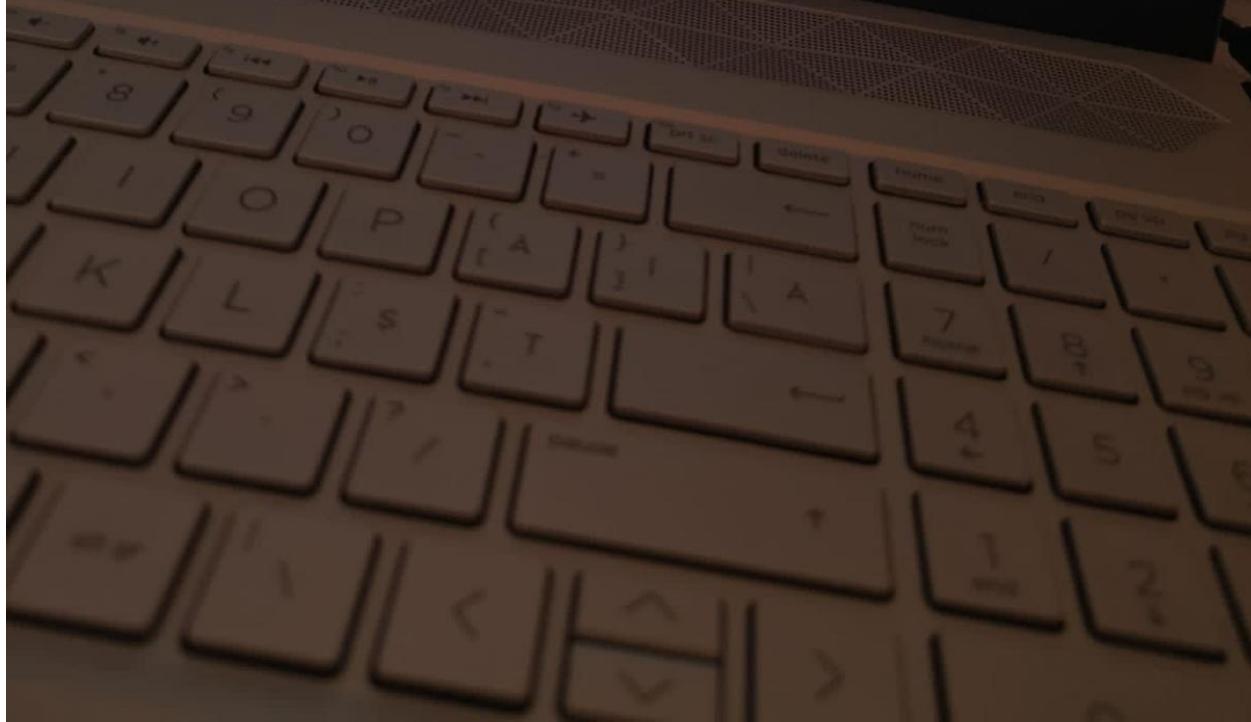
```
class A {
 public void m() {}
 public static void main(String argv[]) {
 A[] a = new A[2];
 /*AICI*/
 }
}
```

Selectați unul sau mai multe:

| a[0] = new A();

| a[0].m();

3°C Cloudy 13:01 ENG 04/02/2022



Selectați unul sau mai multe:

- c = b;
- A a = new A();
- c = d;
- A a = d;
- d = c;

R: a,b,e

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare de compilare când sunt inserate la poziția indicată cu /\*AICI\*/?

```
abstract class A {}
class B extends A {}
class C extends A {}
class D extends C {}
class Main {
 public static void main(String argv[]) {
 B b = new B();
 C c = new C();
 D d = new D();
 /*AICI*/
 }
}
```

Selectați unul sau mai multe:

- d = c;

Întrebare  
a primit  
spuns încă  
arcat din 1,00  
Întrebare cu  
sg

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos, să selectezi cele care pot fi utilizate în cadrul unei expresii de tipul `n(x)`.

```
class A {
 private int a = 0;
 private static int b = 0;

 public int m() {
 return a;
 }

 public static int n(A x) {
 /*AICI*/
 }
}
```

**R: a,d,e**

Selectați unul sau mai multe:

| return m();

| return b;

| return x.a;

| return this.a;

| return a;

Pagina precedentă



Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce

```
class G<T> {
 /* ... */
 void met(T t) /* ... */}
}
class A /* ... */
class B extends A /* ... */
public class Main {
 public static void test(G<? extends A> param) {
 G<A> x = new G<A>();
 G y = new G();
 G<? extends A> z = param;
 /*AICI*/
 }
}
```

**R: a,b,d**

Selectați unul sau mai multe:

z.met(new A());

z.met(new B());

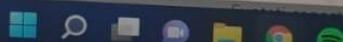
z = x;

x = y;

x.met(new B());

z = y;

Antă



\*/AICI\*/  
}

Selectați unul sau mai multe:

c.equals(a);

c.r(1);

a.q(1);

a.p(1);

a.m(1);

**R: a.q(1)  
a.p(1)**

ASUS

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma execuției programului de mai jos?

```
class A {
 public int m(A p) { return 12; }
 public int n(A p) { return 38; }
}
class B extends A {
 public int m(B p) { return 3;}
 public int n(A p) { return 7; }
}
class C extends A {
 public int m(A p) { return 38; }
 public int n(A p) { return 4; }
}
class D extends A {
 public int m(A p) { return 1; }
 public int n(D p) { return 2; }
}
class Main {
 public static void main(String argv[]) {
 A x = new C();
 A y = new D();
 D z = new D();
 System.out.println(x.m(z) + y.n(z));/*
 }
}
```

R: 76



Ce valoare întreagă se va tipări la linia marcată cu /\* în urma execuției programului de mai jos?

```
class Ex extends RuntimeException {}
class Main {
 public static void m(int x) throws Ex {
 if(x == 2) throw new Ex();
 }
 public static void main(String argv[]) {
 int a = 2;
 try {
 for(int i = 0; i < 4; i++) {
 m(i);
 a = a + 2;
 }
 } catch(Ex e) {
 a++;
 } finally {
 a++;
 }
 System.out.println(a);/*
 }
}
```

Răspuns:

8

ASUS

Considerând următorul cod Java, care din fragmentele de cod enumerate mai jos vor produce o eroare sunt inserate la poziția indicată cu /\*AICI\*/?

```
abstract class A {
 public void m(int p) {}
 public void r(int p) {}
}

class B extends A {
 public void m(int p) {}
 public void p(int p) {}
 public void q(int p) {}
}

class C extends A {
 public void m(int p) {}
 public void p(int p) {}
 public void r(int p, int t) {}
}

class Main {
 public static void main(String argv[]) {
 A a = new B();
 C c = new C();
 /*AICI*/
 }
}
```



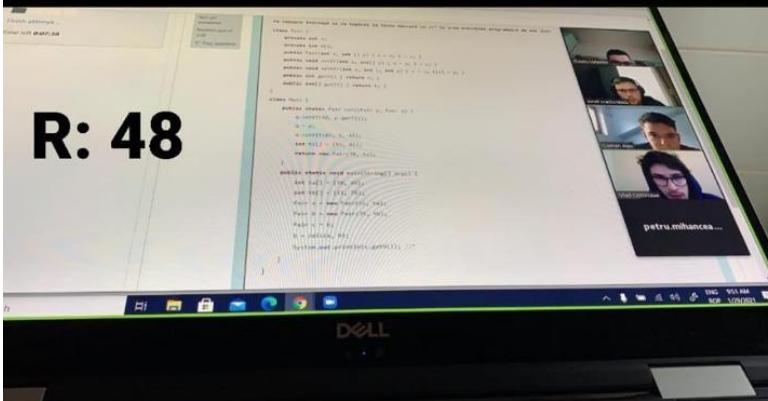
Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programului de mai jos?

```
class A {
 private static int x;
 private int y;
 public A(int x1, int y1) { x = x1; y = y1;}
 public void setXY(int x1, int y1) { x = x1; y = y1;}
 public int getX() { return x; }
 public int getY() { return y; }
}
class Main {
 public static void m(A z, A t) {
 z.setXY(3,20);
 z = new A(2,17);
 }
 public static void main(String[] args) {
 A a = new A(9,50);
 A b = new A(7,28);
 m(a,b);
 System.out.println(a.getY() + b.getX());/*
 }
}
```

Răspuns:

**22**





21:28

85%



De valoare întragă se va tipări la linia de mai jos?

```
class X {
 protected int v = 0;
 public X() { v += 87; }
}

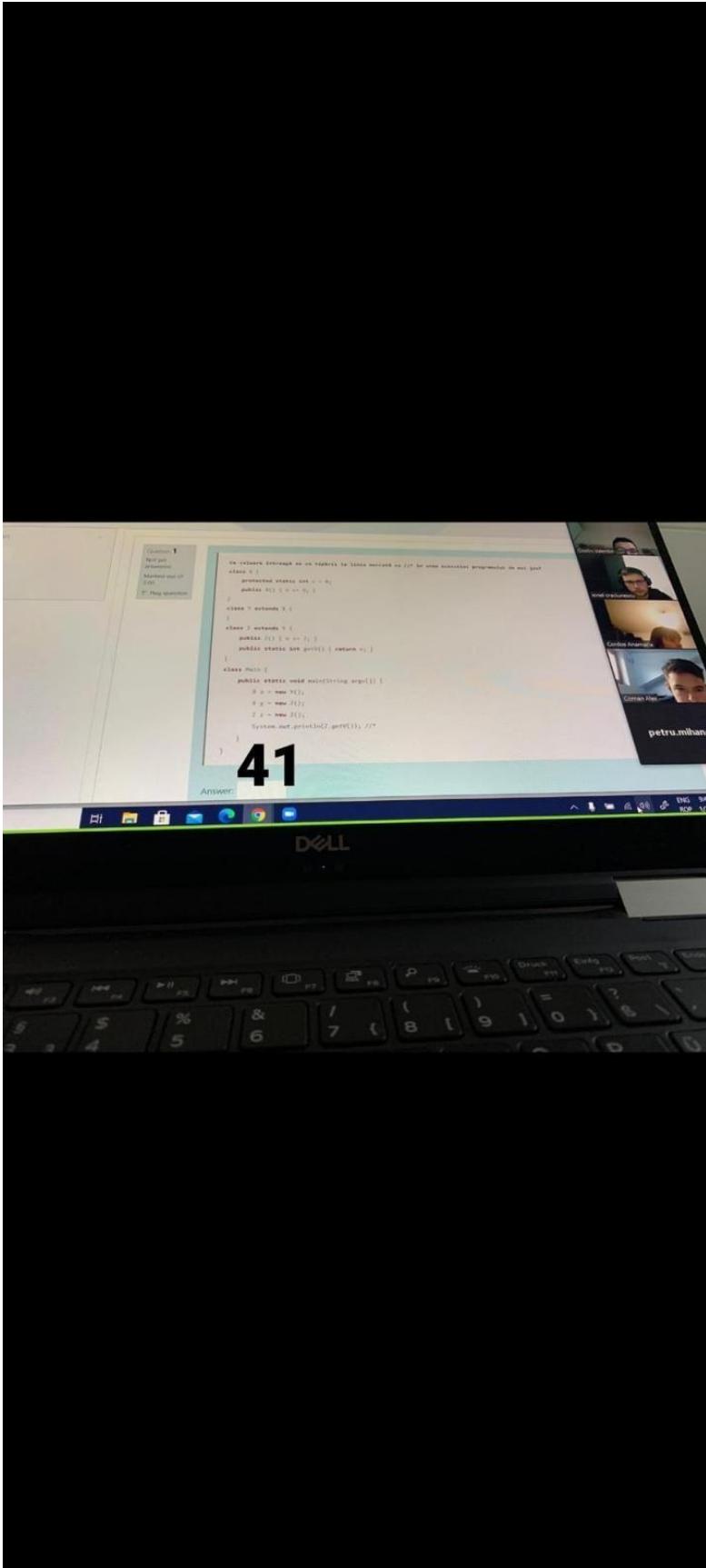
class Y extends X {
 public Y() { v += 94; }
 public int getV() { return v; }
}

class Z extends Y {
 public Z() { v += 46; }
}

class Main {
 public static void main(String argv[]) {
 X x = new Z();
 Y y = new Y();
 Z z = new Z();
 System.out.println(y.getV()); /* */
 }
}
```

Answer: **181**





Not yet answered  
Marked out of 3.00  
Flag question

Ce valoare întregă se va tipări la linia marcată cu `/*` în urma executiei programului de mai jos?

```
class Pair {
 private int v;
 private int t[];
 public Pair(int x, int [] y) { v = x; t = y; }
 public void setV(int x, int[] y) { v = x; t = y; }
 public void setVII(int x, int i, int y) { v = x; t[i] = y; }
 public int getV() { return v; }
 public int[] getT() { return t; }
}
class Main {
 public static Pair call(Pair p, Pair q) {
 q.setV(100, p.getT());
 q = p;
 q.setVII(53, 1, 34);
 int t2[] = {88, 81};
 return new Pair(20, t2);
 }
 public static void main(String[] args) {
 int ta[] = {71, 59};
 int tb[] = {11, 61};
 Pair a = new Pair(19, ta);
 Pair b = new Pair(83, tb);
 Pair c = b;
 b = call(a, b);
 System.out.println(c.getT()[1]); /*
 }
}
```

**34**

Question 3

Not yet  
answered

Marked out of  
3.00

Flag question

Ce valoare întreagă se va tipări la linia marcată cu ///\* în urma executiei programării?

```
class ExA extends Exception {}
class ExB extends ExA {}
class Main {
 public static int make(int a) throws ExA {
 for (int k = 0; k < 3; k++) {
 a = a + 2;
 if (a == 3) throw new ExB();
 }
 return a;
 }
 public static void main(String argv[]) throws ExA {
 int x = 0;
 for (int i = 0; i < 2; i++) {
 try {
 x++;
 x = make(x);
 x++;
 } catch (ExB e) {
 System.out.println(e);
 } finally {
 x++;
 }
 }
 System.out.println(x); ///*
 }
}
```

Answer: 11

## Înțeles Obiecte

Question 4

Not yet  
answered  
Marked out of  
1.00  
Flag question

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programării?

```
abstract class A {
 public int proc(B p) { return 25; }
}

class B extends A {
 public int proc(C p) { return 96;}
}

class C extends B {
 public int proc(C p) { return 99; }
}

public class Main {
 public static void main(String argv[]) {
 A x = new C();
 C y = new C();
 C z = new C();
 System.out.println(x.proc(z) + y.proc(z)); /*
 }
}
```

Answer: 124

Previous page

GET IT ON

Download on the

568&cmid=88563&page=1

on 2  
ed  
put of  
question

```
Ce valoare întreagă se va tipării la linia marcată cu //* în urma execuției programului de
sus? (încălzirea săptămânii)
```

```
class Pair {
 private int v;
 private static int t[];
 public Pair(int x, int[] y) { v = x; t = y; }
 public void setV(int x, int[] y) { v = x; t = y; }
 public void setVIT(int x, int i, int y) { v = x; t[i] = y; }
 public int getV() { return v; }
 public int[] getT() { return t; }
}
class Main {
 public static Pair call(Pair p, Pair q) {
 p.setV(17, q.getT());
 p = q;
 p.setVIT(72, 1, 21);
 int t[] = {89, 41};
 return new Pair(84, t);
 }
 public static void main(String[] args) {
 int ta[] = {0, 13};
 int tb[] = {45, 99};
 Pair a = new Pair(66, ta);
 Pair b = new Pair(55, tb);
 Pair c = b;
 b = call(a, c);
 System.out.println(b.getT()[0]);
 }
}
```

Answer: **45**

Not yet  
answered

Marked out of  
3.00

Flag question

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programului?

```
class Ex extends Exception {}
class Main {
 public static void main(String argv[]) throws Ex {
 int i = 0;
 int z = 0;
 while (i < 2) {
 try {
 int k = 0;
 while (k < 3) {
 k++;
 z = z + 2;
 if (i == 0) throw new Ex();
 }
 z = z + 2;
 } catch (Ex e) {
 System.out.println(e);
 } finally {
 z = z + 3;
 }
 i++;
 }
 System.out.println(z); /*
 }
}
```

12

Answer:

← 9.jpg



Question 1  
Not yet  
answered  
Marked out of  
3.00  
 Flag question

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programului de mai jos?

```
abstract class A {
 public int proc(A p) { return 20; }
}
class B extends A {
 public int proc(A p) { return 100; }
}
class C extends A {
 public int proc(C p) { return 75; }
}
public class Main {
 public static void main(String argv[]) {
 C x = new C();
 A y = new B();
 C z = new C();
 System.out.println(y.proc(x) + z.proc(x)); /*
 }
}
```

Answer

175



21:25

86%

← 8.jpg



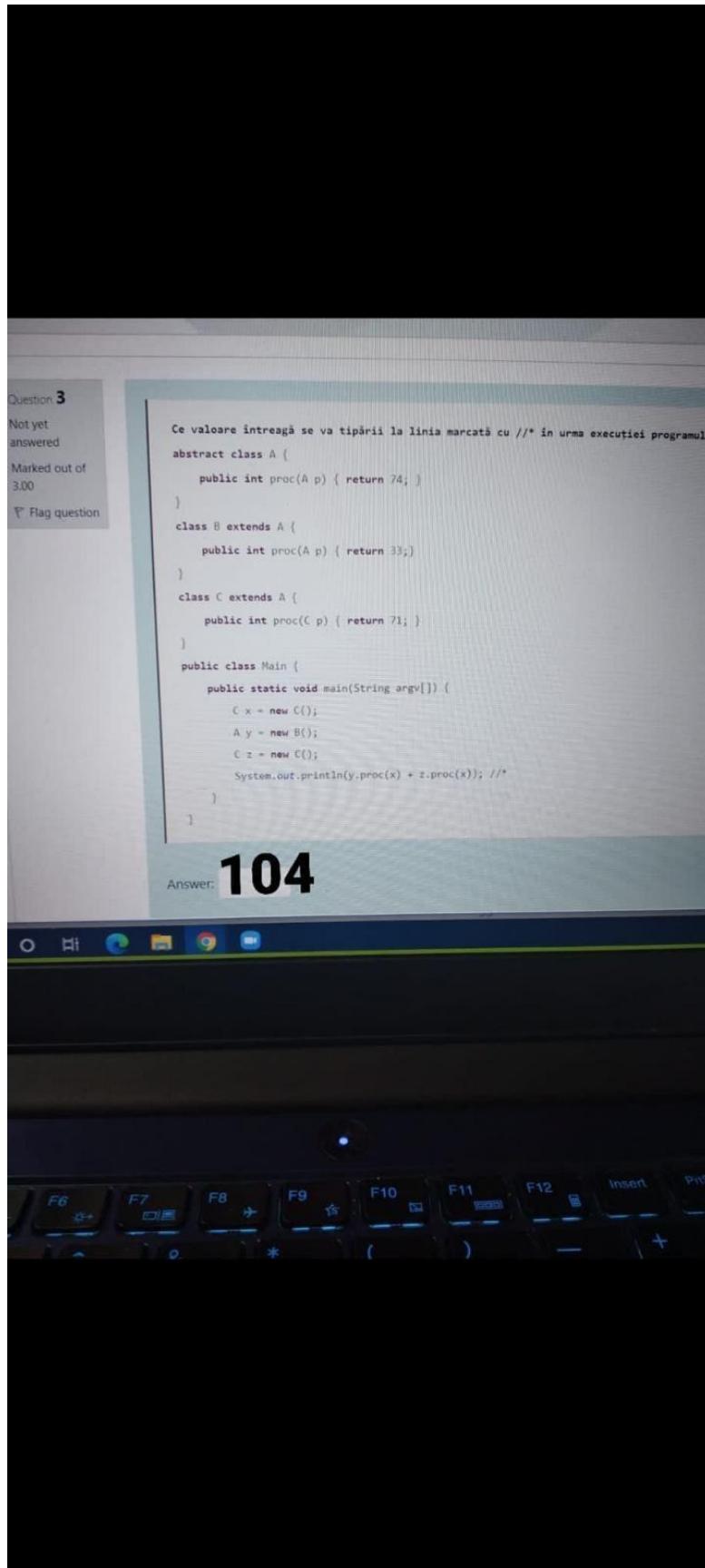
answered  
Marked out of  
3.00  
Flag question

R: 87

```
class Pair {
 private int v;
 private int t[];
 public Pair(int x, int[] y) { v = x; t = y; }
 public void setVT(int x, int[] y) { v = x; t = y; }
 public void setVIT(int x, int i, int y) { v = x; t[i] = y; }
 public int getV() { return v; }
 public int[] getT() { return t; }
}
class Main {
 public static Pair call(Pair p, Pair q) {
 p.setVT(76, q.getT());
 p = q;
 p.setVIT(87, 0, 74);
 int tz[] = {97, 93};
 return new Pair(19, tz);
 }
 public static void main(String[] args) {
 int ta[] = {29, 35};
 int tb[] = {6, 87};
 Pair a = new Pair(3, ta);
 Pair b = new Pair(50, tb);
 Pair c = b;
 b = call(a, c);
 System.out.println(c.getV()); /*
 }
}
```

După apelare:  
a={v=76, t[]=[74, 87]}  
b={v=19, t[]=[97, 93]}  
c={v=87, t[]=[74, 87]}





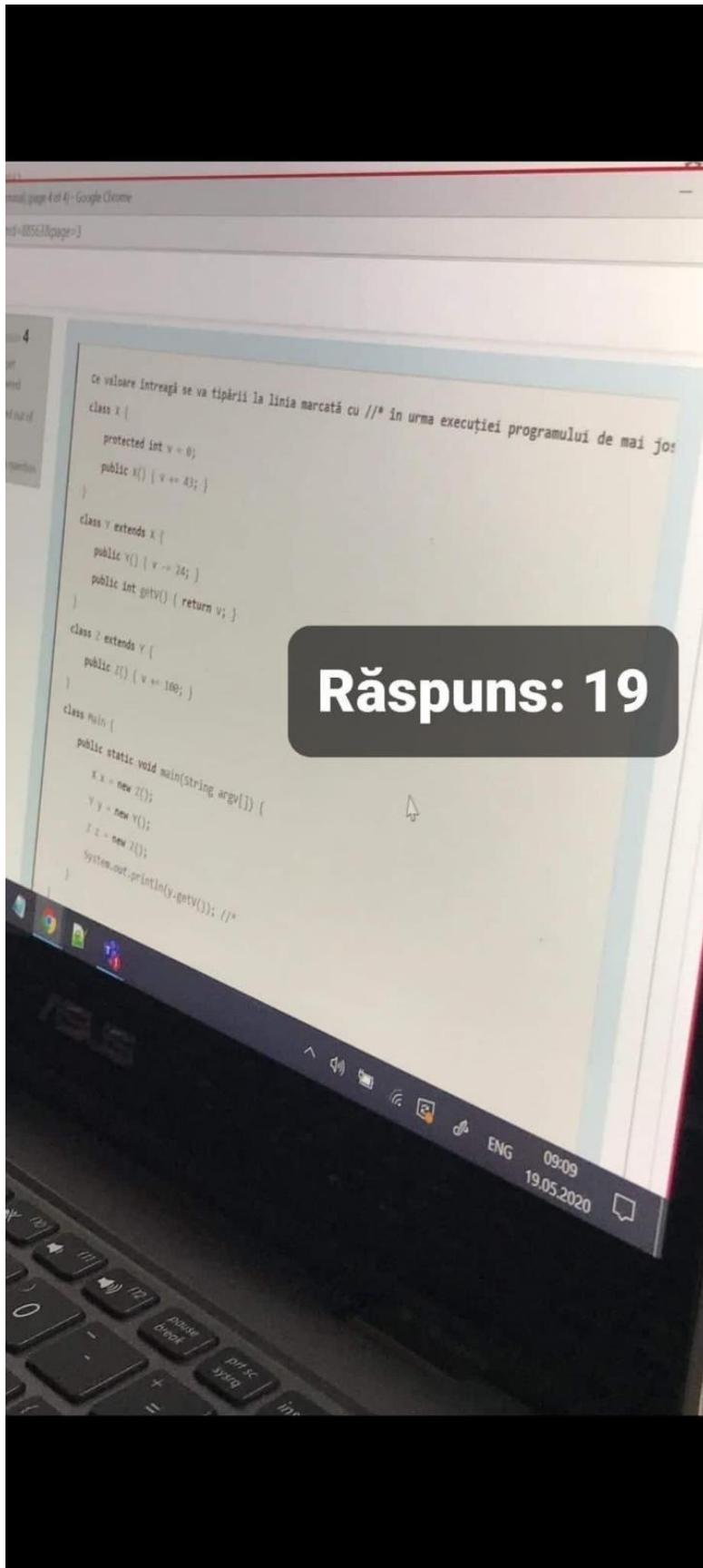
**Răspuns: 19**

```
4
Ce valoare întreagă se va tipări la linia marcată cu /* în urma execuției programului de mai jos
class X {
 protected int v = 9;
 public X() { v += 43; }
}

class Y extends X {
 public Y() { v -= 24; }
 public int getv() { return v; }
}

class Z extends Y {
 public Z() { v += 109; }
}

class Main {
 public static void main(String args[]) {
 X x = new Z();
 Y y = new Y();
 Z z = new Z();
 System.out.println(y.getv()); /*
```



21:25

86%

← 7.jpg



Answered

Marked out of  
3.00

Flag question

R: 8

```
class Ex extends Exception {}

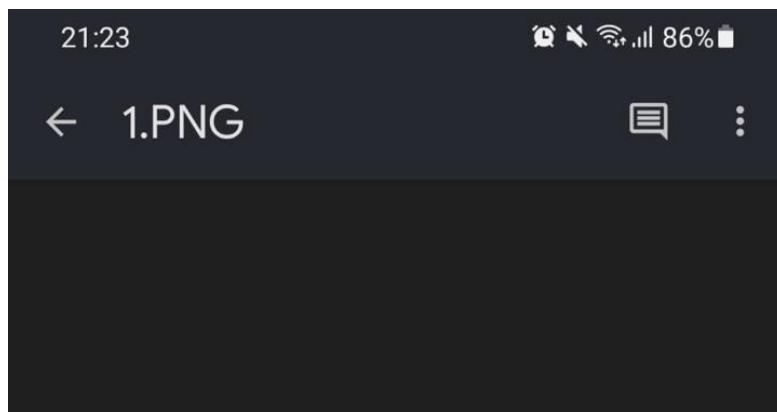
class Main {
 public static void main(String argv[]) throws Ex {
 int i = 0;
 int z = 0;
 while (i < 2) {
 try {
 int k = 0;
 while (k < 1) {
 k++;
 z = z + 2;
 if (i == 0) throw new Ex();
 }
 z = z + 2;
 } catch (Ex e) {
 System.out.println(e);
 } finally {
 z = z + 1;
 }
 i++;
 }
 System.out.println(z); /*
 }
}
```

Se afișează:

Ex

8





Ce valoare întreagă se va tipări la linia marcată cu /\* în urma execuției programului de mai jos:

```
abstract class A
{
 public int proc(A p)
 {
 return 98;
 }
}

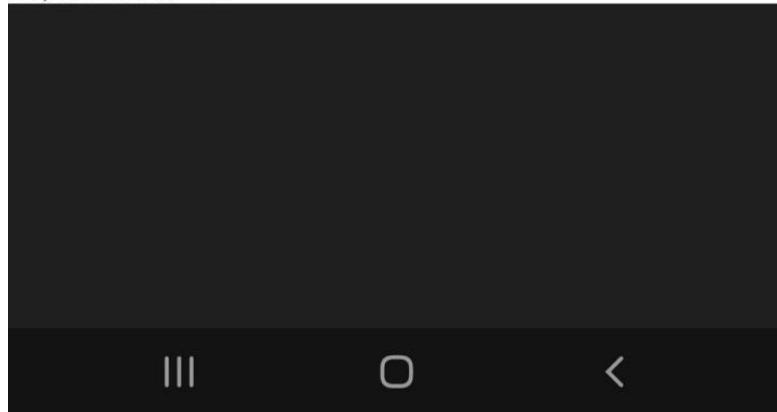
class B extends A
{
 public int proc(A p)
 {
 return 17;
 }
}

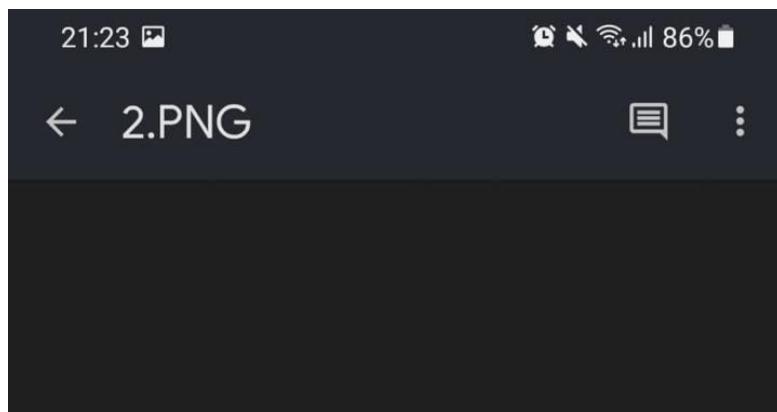
class C extends A
{
 public int proc(C p)
 {
 return 65;
 }
}

public class Main
{
 public static void main(String argv[])
 {
 C x = new C();
 A y = new B();
 C z = new C();
 System.out.println(y.proc(x) + z.proc(x)); /*
 }
}
```

**R: 82**

Răspuns: 17 + 65 = 82





Ce valoare întreagă se va tipări la linia marcată cu `/*` în urma execuției programului de mai jos:

```
abstract class A
{
 public int proc(B p)
 {
 return 5;
 }
}

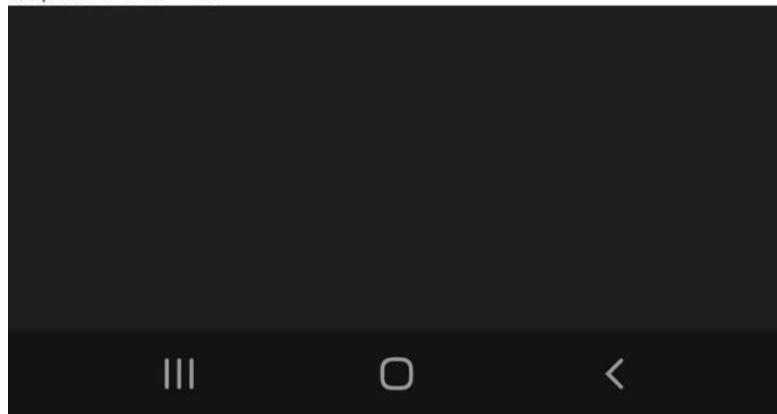
class B extends A
{
 public int proc(C p)
 {
 return 21;
 }
}

class C extends B
{
 public int proc(C p)
 {
 return 45;
 }
}

public class Main
{
 public static void main(String argv[])
 {
 A x = new C();
 C y = new C();
 C z = new C();
 System.out.println(x.proc(z) + y.proc(z)); /*
 }
}
```

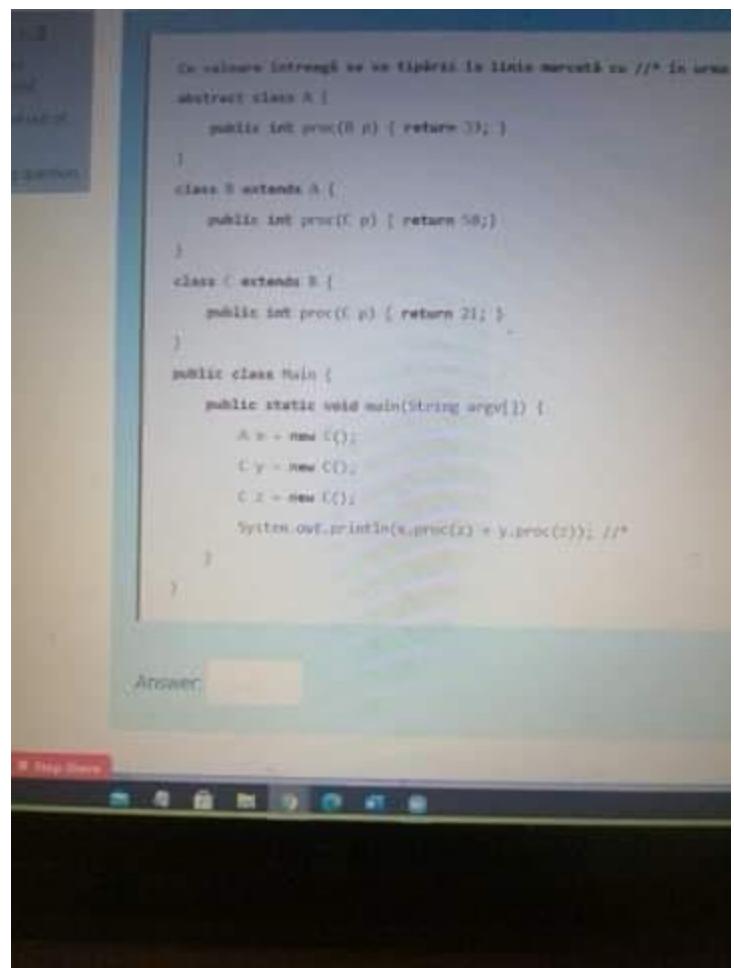
**R: 50**

Răspuns:  $5 + 45 = 50$



```
de valoare întreagă se va tipări în linia menită cu //* în urmă
abstract class A {
 public int proc(B p) { return 33; }
}
class B extends A {
 public int proc(C p) { return 56; }
}
class C extends B {
 public int proc(D p) { return 21; }
}
public class Main {
 public static void main(String args[]) {
 A x = new C();
 C y = new C();
 C z = new C();
 System.out.println(x.proc(y) + y.proc(z)); /*
 }
}
```

Answer:



```
class Test extends IOException {
 class Test extends IOException {}
 class Test extends IOException {}
 class Test {
 public static int mahr(int x) throws IOException {
 for (int i = 0; i < 2; i++) {
 x = x - 1;
 if (x == -2) throw new IOException();
 }
 return x;
 }
 public static void main(String args[]) throws IOException {
 int x = 0;
 for (int i = 0; i < 2; i++) {
 try {
 x = mahr(x);
 x = x - 1;
 } catch (IOException e) {
 System.out.println(e);
 } finally {
 x = x - 1;
 }
 }
 System.out.println(x); // -1
 }
 }
}
```

```
class Pair {
 private int x1;
 private int t1;
 public Pair(int x, int t) { x = x1; t = t1; }
 public void setV1(int x, int t) { x = x1; t = t1; }
 public void setV2(int x, int t) { x = x2; t = t2; }
 public int getV1() { return x1; }
 public int getV2() { return t1; }
}
class Main {
 public static Pair call(Pair p, Pair q) {
 p.setV1(15, q.getV2());
 p = q;
 p.setV1(34, 0);
 int tr[] = {56, 58};
 return new Pair(62, t2);
 }
 public static void main(String[] args) {
 int ta[] = {69, 56};
 int tb[] = {16, 11};
 Pair a = new Pair(73, ta);
 Pair b = new Pair(91, tb);
 Pair c = b;
 b = call(a, c);
 System.out.println(c.getV1());
 }
}
```

Question 3

Not yet  
answered

Marked out of  
1.00

Flag question

Ce valoare întreagă se va tipări în linia marcată cu /\* în urma executiei programului?

```
class ExA extends Exception {}
class ExB extends ExA {}

class Main {
 public static int make(int a) throws ExA {
 for (int k = 0; k < 3; k++) {
 a = a + 2;
 if (a == 3) throw new ExB();
 }
 return a;
 }

 public static void main(String argv[]) throws ExA {
 int x = 0;
 for (int i = 0; i < 2; i++) {
 try {
 x++;
 x = make(x);
 x++;
 } catch (ExB e) {
 System.out.println(e);
 } finally {
 x++;
 }
 }
 System.out.println(x); // /*
 }
}
```

Answer: 11

Ce valoare întreagă se va tipări la linia marcată cu /\* în urma executiei programului de mai jos?

```
class A {
 private static int x;
 private int y;
 public A(int x1, int y1) { x = x1; y = y1;}
 public void setXY(int x1, int y1) { x = x1; y = y1;}
 public int getX() { return x; }
 public int getY() { return y; }
}
class Main {
 public static void m(A z, A t) {
 z.setXY(3,20);
 z = new A(2,17);
 }
 public static void main(String[] args) {
 A a = new A(9,50);
 A b = new A(7,28);
 m(a,b);
 System.out.println(a.getY() + b.getX());/*
 }
}
```

Răspuns:

**22**



Ce valoare întreagă se va tipări la linia marcată cu /\* în urma execuției programului de mai jos?

```
class Ex extends RuntimeException {}
class Main {
 public static void m(int x) throws Ex {
 if(x == 2) throw new Ex();
 }
 public static void main(String argv[]) {
 int a = 2;
 try {
 for(int i = 0; i < 4; i++) {
 m(i);
 a = a + 2;
 }
 } catch(Ex e) {
 a++;
 } finally {
 a++;
 }
 System.out.println(a);/*
 }
}
```

Răspuns:

8



prt sc

```
1.|
class A{
 public static int x = 0;

 public A(){
 x+=3;
 }

 public static int show(){
 return x;
 }
}

class B extends A{
 public B(){
 x+=1;
 }
}

class C extends A{
 public C(){
 x += 3;
 }
}
public class Main
{
 public static void main(String[] args) {
 A a = new C();
 B b = new B();
 C c = new C();
 System.out.println(A.show());
 }
}
```

→ output : 16 ( aveti grija la modificadorul STATIC, va fi diferit, se aduna toate)

---

2.

```
interface I{}
interface J{}
interface IJ extends I, J{}
```

```
class A implements I{}
class B extends A implements J{}
class C extends A{}
class D implements IJ{}
```

```
class Main{
 public void doSomething(I i, J j, IJ jj){
 A a = new A();
 B b = new B();
 C c = new C();
 D d = new D();
 }
}
```

→ i = jj, jj = j, i = c, j = c, b = d;

3. Scrie raspunsul :

```
class Ex1 extends Exception{}
class Ex2 extends Ex1{}
```

```
class Main{
 public static void main(String []args){
 int i = 0;
 int a = 9;
 ok
 while(i < 2){
 try{
 int k = 0;
 while(k < 2){
 k++;
 a++;
 if(i == 0) throw new Ex2();
 }
 a++;
 }catch(Ex2 e){
 a = a + 1;
 }catch(Exception e){
 a += 2;
 }finally{
 a++;
 }
 i++;
 }

 System.out.println(a);/
 }
}
```



4.

```
class Ex1 extends Exception{}
class Ex2 extends Ex1{}

class Main{
 public static int m(int x) throws Ex1{
 for(int k = 0; k < 1; k++){
 x += 2;
 if(x == 3){
 throw new Ex1();
 }
 }
 return x;
 }
 public static void main(String []argv){
 int a = 0;

 for(int i = 0; i < 2; i++){
 try{
 a++;
 a = m(a);
 a++;
 }catch(Ex2 e){
 a += 1;■
 }catch(Ex1 e){
 a += 2;
 }finally{
 a++;
 }
 }

 System.out.println(a);/
 }
}
```

5. Care din variantele de mai jos introduse in linia marcata cu //AICI va da eroare de compilare?

```
interface I {
 public void m();
}
interface J {
 public void n();
}
class A extends I {
 public void m();
}
class B extends J {
 public void n();
}
//AICI
|
class C extends A implements J { // corect
 public void m();
 public void n();
}
class C extends B implements I { //corect
 public void m();
}

class C extends A, B { //eroare compilare
 public void p();
}

class C implements I { //eroare compilare
 public abstract n();
}
class C extends A implements J { //eroare compilare
 public void p();
}
```

6. Care din urmatoarele sechete de cod introduce la linia marcată cu //AICI vor da eroare de compilare?

```
class Ex1 extends Exception {}
class Ex2 extends Exception {}
class Ex3 extends Ex1 {}
class Ex4 extends RuntimeException {}
class Ex5 extends Ex3 {}
class Ex6 extends Ex4 {}
class Ex7 extends Ex2 {}
class Ex8 extends Ex2 {}

class Main {
 public static void make (int x) throws Ex3 {
 if (x == 2)
 throw new Ex5();
 }
 public static void main (String[] args) {
 try {
 for (int i = 0; i < 3; i++) {}
 make(i);
 //AICI
 } }
a)
catch (Ex3 e){ //corect
}
b)
catch (Ex5 e) { //corect

} catch (Ex1 e) {}
c)
catch (Ex3 e) { //eroare
} catch (Ex5 e) {
}
d)
catch (Ex1 e) { //corect
}
e)
catch (Ex5 e) { //eroare
}
```