

Started on Monday, 3 April 2023, 9:00 AM

State Finished

Completed on Monday, 3 April 2023, 11:30 AM

Time taken 2 hours 29 mins

Grade 66.02 out of 100.00

Question **1**

Correct

Mark 2.00 out of 2.00

Yang akan terjadi Error pada penamaan Variabel berikut adalah?

Select one:

- ☒ a. 1Nama ✓
- ☐ b. \$Nama
- ☐ c. Nama1
- ☐ d. _Nama

Your answer is correct.

The correct answer is: 1Nama

Question **2**

Incorrect

Mark 0.00 out of 2.00

Statement berikut yang akan menyebabkan error ketika program dijalankan adalah?

Select one:

- ☐ a. `int s = (int) 10.2 + 0.8;`
- ☐ b. `float f = 12.75f;`
- ☒ c. `byte b = (byte) 135;` ✗
- ☐ d. `long myVar = 123_456_789L;`

Your answer is incorrect.

The correct answer is: `int s = (int) 10.2 + 0.8;`

Question **3**

Correct

Mark 2.00 out of 2.00

Nilai Default untuk tipe data int adalah?

Select one:

- ☒ a. 0 ✓
- ☐ b. Null
- ☐ c. False
- ☐ d. True

Your answer is correct.

The correct answer is: 0

Question **4**

Correct

Mark 2.00 out of 2.00

Pilih **True** or **False**.

apakah Ouput pada program berikut berupa

Lampu merah, berhenti!

Lampu kuning, harap hati-hati!

Lampu hijau, silahkan jalan!

```
public class LampuLalulintas {  
    public static void main(String[] args) {  
  
        String lampu ="merah";  
  
        switch(lampu){  
            case "merah":  
                System.out.println("Lampu merah, berhenti!");  
            case "kuning":  
                System.out.println("Lampu kuning, harap hati-hati!");  
            case "hijau":  
                System.out.println("Lampu hijau, silahkan jalan!");  
                break;  
            default:  
                System.out.println("Warna lampu salah!");  
        }  
    }  
}
```

Select one:

- ☒ True ✓
- ☐ False

Lampu merah, berhenti!

Lampu kuning, harap hati-hati!

Lampu hijau, silahkan jalan!

The correct answer is 'True'.

Apa output dari program berikut ini ?

```
public class Test3{  
    public static void main(String[] args){  
        mystery(7);  
    }  
  
    public static void mystery(int n){  
        if(n<=0){  
        }else{  
            mystery(n-2);  
            System.out.println(n/2);  
        }  
    }  
}
```

Select one:

☒ a. Output: ✓

0
1
2
3

☐ b. Output:

3
2
1
0

☐ c. Output:

3
2
1

☐ d. Output:

1
2
3

Your answer is correct.

The correct answer is: Output:

0
1
2
3

Question 6

Incorrect

Mark 0.00 out of
2.00

Apa yang akan dihasilkan jika program berikut ini dijalankan?

```
public class Test4{  
    public static void main(String[] args){  
        areaCircle(10);  
    }  
  
    public double areaCircle(int x){  
        return Math.PI * Math.pow(x,2);  
    }  
}
```

Select one:

- ☐ a. Program tidak dapat dijalankan karena method areaCircle tidak dapat dipanggil secara langsung dari method main
- ☐ b. Program tidak dapat dijalankan karena class Math dalam package java.lang belum diimport
- ☐ c. Program tidak dapat dijalankan karena method pow dalam kelas Math hanya menerima input bertipe double
- ☒ d. Program dapat dijalankan dengan baik dan memberikan output yang benar ✖

Your answer is incorrect.

The correct answer is: Program tidak dapat dijalankan karena method areaCircle tidak dapat dipanggil secara langsung dari method main

Diketahui terdapat method berikut ini dalam file `Programku.java`:

```
public static void test (int x) {  
    // do something  
}
```

Tentukan **semua method** dalam tabel berikut ini yang jika ditambahkan ke dalam file `ProgramKu.java` akan menyebabkan error!

Nomor	Method
1	<pre>public static void test(int y){ //do something }</pre>
2	<pre>public static void test(int x, int y){ // do something }</pre>
3	<pre>public static int test (int a) { //do something }</pre>
4	<pre>public static void test(){ //do something }</pre>
5	<pre>public static void test(double a){ //do something }</pre>

Select one:

- ☐ a. method no. 2, 4, 5
- ☒ b. method no. 1, 3 ✓
- ☐ c. method no. 1
- ☐ d. method no. 1, 3, 5

Your answer is correct.

The correct answer is: method no. 1, 3

Question 8

Incorrect

Mark 0.00 out of 2.00

Method recursive `div` berikut ini akan melakukan pembagian dua buah bilangan bulat, yaitu $x \div y$. Asumsi parameter `x` dan `y` yang di-passing pada pemanggilan pertama method `div` (bukan recursive call) selalu merupakan bilangan bulat positif. Manakah base case yang tepat untuk program berikut ini?

```
// asumsi parameter x dan y yang di-passing pada pemanggilan pertama method div (bukan recursive call)
public static int div(int x, int y){

    //lengkapi base case method recursive ini
    _____;
    //recursion case
    else return 1+div(x-y, y);

}
```

Select one:

- ☐ a. `if(x<=y) return 0;`
- ☒ b. `if(x<=0) return 0;` ✖
- ☐ c. `if(x==0) return 0;`
- ☐ d. `if(x<y) return 0;`

Your answer is incorrect.

The correct answer is: `if(x<y) return 0;`

Question 9

Incorrect

Mark 0.00 out of 2.00

Perhatikan potongan kode di bawah

```
int[] arr = new int[5];
arr = new int[6];
```

Manakah pernyataan di bawah ini yang paling sesuai?

Select one:

- ☐ a. Kode berjalan dengan baik tanpa *error*.
- ☒ b. Terjadi compile error karena tidak bisa meng-*assign array* dengan ukuran berbeda pada suatu variabel. ✖
- ☐ c. Terjadi *runtime error* karena variabel **arr** tidak bisa diubah setelah di *assign* suatu *object array*.
- ☐ d. Terjadi *compile error* karena variabel **arr** tidak bisa diubah setelah di *assign* suatu *object array*.

Your answer is incorrect.

The correct answer is: Kode berjalan dengan baik tanpa *error*.

Question **10**

Incorrect

Mark 0.00 out of 2.00

Berapakah nilai a[1] setelah potongan kode di bawah dieksekusi

```
int[] a = {0,2,4,1,3};  
for(int i = 0; i < a.length; i++){  
    a[i] = a[(a[i] + 3) % a.length];  
}
```

Select one:

- ☐ a. 0
- ☐ b. 1
- ☒ c. 2 ✖
- ☐ d. 3

Your answer is incorrect.

The correct answer is: 1

Question **11**

Correct

Mark 2.00 out of 2.00

Ada berapa statement deklarasi array di bawah ini yang legal untuk dilakukan?

```
[]double num1;  
double[] num;  
double num[];
```

Select one:

- ☒ a. Dua ✔
- ☐ b. Tidak ada
- ☐ c. Satu
- ☐ d. Tiga

Your answer is correct.

The correct answer is: Dua

Question **12**

Incorrect

Mark 0.00 out of 2.00

Ketika sebuah *object array* di-pass melalui pemanggilan suatu *method*, apa yang diterima oleh *method* tersebut?

Select one:

- ☒ a. copy dari array ✖
- ☐ b. copy dari elemen pertama array
- ☐ c. reference dari array
- ☐ d. panjang dari array

Your answer is incorrect.

The correct answer is: reference dari array

Question **13**

Correct

Mark 2.00 out of 2.00

Perhatikan potongan code berikut ini. Asumsikan bahwa `Car` adalah class yang sudah didefinisikan yang memiliki constructor default.

```
Car[] myCars= new Car[3];  
myCars[0] = new Car();
```

Di akhir eksekusi potongan code ini, `myCars[0]` dan `myCars[1]` masing-masing akan berisi...

Select one:

- ☐ a. `null` dan `null`
- ☐ b. Reference kepada array `myCars` dan `null`
- ☐ c. Object `Car` dan `null`
- ☒ d. Reference kepada object `Car` dan `null` ✓

Your answer is correct.

The correct answer is: Reference kepada object `Car` dan `null`

Question **14**

Correct

Mark 2.00 out of 2.00

Perhatikan class `Handphone` berikut ini. Jika statement saya jalankan dari main di class yang berbeda, statement yang manakah yang **TIDAK** akan menimbulkan error?

```
class Handphone{  
    private String brand;  
    private double price;  
  
    public void turnOn(){  
        System.out.println("Phone on!")  
    }  
}
```

Select one:

- ☒ a. `Handphone hpku = new Handphone();` ✓
`System.out.println(hpku);`
- ☐ b. `Handphone hpku = new Handphone();`
`Handphone.turnOn();`
- ☐ c. `Handphone hpku = new Handphone("Sumsang", 9000000);`
`hpku.turnOn();`
- ☐ d. `Handphone hpku = new Handphone();`
`System.out.println(hpku.price);`

Your answer is correct.

The correct answer is: `Handphone hpku = new Handphone();`
`System.out.println(hpku);`

Question **15**

Correct

Mark 2.00 out of 2.00

Perhatikan class `Obat` berikut ini. Pernyataan manakah yang BENAR?

```
class Obat{  
    private double volume;  
}
```

Select one:

- ☐ a. Object jenis `Obat` tidak dapat dibuat karena volume tidak diinisialisasi
- ☐ b. Object jenis `Obat` tidak dapat dibuat karena tidak ada constructor
- ☒ c. Object jenis `Obat` dapat diinstansiasi ✓
- ☐ d. Object jenis `Obat` tidak dapat dibuat karena tidak bisa menentukan volume

Your answer is correct.

The correct answer is: Object jenis `Obat` dapat diinstansiasi

Question **16**

Correct

Mark 2.00 out of 2.00

Perhatikan potongan kode berikut ini.

```
1  int a = 1;  
2  int b = 2;  
3  double c = 4.2;  
4  Integer d = b;  
5  Double e = c;  
6  c = b;  
7  a = d;  
8  b = a;
```

Baris yang menunjukkan autoboxing dan unboxing adalah:

Select one:

- ☐ a. Baris 6 dan baris 8
- ☒ b. Baris 5 dan baris 7 ✓
- ☐ c. Baris 1 dan baris 4
- ☐ d. Baris 3 dan baris 4

Your answer is correct.

The correct answer is: Baris 5 dan baris 7

Perhatikan program berikut (terdapat beberapa berkas Java). Apa outputnya jika program tersebut dijalankan?

```
// Kendaraan.java
public class Kendaraan {

    public void klakson() {
        klakson("Tintin");
    }

    public void klakson(String str) {
        System.out.println(str);
    }

}

// Bus.java
public class Bus extends Kendaraan {

    public void klakson(){
        klakson("Telolet");
    }

}

// TesKlakson.java
public class TesKlakson {

    public static void main(String[] args) {
        Kendaraan bus = new Kendaraan();
        bus.klakson();
    }

}
```

Select one:

- ☒ Tintin ✓
- ☐ Telolet
- ☐ Bus
- ☐ Terjadi error

Your answer is correct.

The correct answer is: Tintin

Question **18**

Correct

Mark 2.00 out of 2.00

Perhatikan program berikut (terdapat beberapa berkas Java). Hubungan antara method klakson() pada class Bus dengan method klakson() pada class Kendaraan disebut dengan ...

```
// Kendaraan.java
public class Kendaraan {

    public void klakson() {
        klakson("Tintin");
    }

    public void klakson(String str) {
        System.out.println(str);
    }

}

// Bus.java
public class Bus extends Kendaraan {

    public void klakson(){
        klakson("Telolet");
    }

}
```

Select one:

- ☒ overriding ✓
- ☐ interfacing
- ☐ overrating
- ☐ subclassing

Your answer is correct.

The correct answer is: overriding

Question **19**

Incorrect

Mark 0.00 out of 2.00

Manakah yang SALAH tentang access modifier "protected" pada Java?

Select one:

- ☒ protected field dapat diakses dari subclass di package yang sama ✗
- ☐ protected field dapat diakses dari superclass untuk kelas yang mengandung deklarasi field tersebut
- ☐ protected field dapat diakses dari kelas yang mengandung deklarasi field tersebut
- ☐ protected field dapat diakses dari subclass di package yang berbeda

Your answer is incorrect.

The correct answer is: protected field dapat diakses dari superclass untuk kelas yang mengandung deklarasi field tersebut

Question **20**

Correct

Mark 2.00 out of
2.00

Manakah yang merupakan contoh dari multilevel inheritance di Java?

Select one:

- ☐ Class A extends Class B dan Class A extends Class C
- ☐ Class A extends Class B dan Class C extends Class D
- ☐ Class A extends Class B dan Class B extends Class A
- ☒ Class A extends Class B dan Class B extends Class C ✓

Your answer is correct.

The correct answer is: Class A extends Class B dan Class B extends Class C

1. Lengkapi Code yang kurang berikut agar Method Factorial dan Method Factorial1 dapat berjalan dengan baik .

```

1  public class Bismillah {
2
3      public static void main(String[] args) {
4          int a = factorial(3);
5          System.out.print(a);
6          System.out.print("\n");
7
8          int b = factorial1(3,1);
9          System.out.print(b);
10     }
11
12     public static int factorial(int n) {
13         if (n <= 1) {
14             return 1;
15         }
16         else {
17             (--Isi no (1)--)
18         }
19     }
20
21     public static int factorial1(int n, int a) {
22         if (n <= 1) {
23             return a;
24         }
25         else {
26             (--Isi no (2)--)
27         }
28     }
29 }

```

Tuliskan Jawaban Anda pada kolom isian dibawah ini masing-masing 2 point (NOTE: ISI TANPA MENGGUNAKAN SPASI)

(1) (--Isi no (1)--): ✓

(2) (--Isi no (2)--): ✗

Comment:

```
jawaban 1
return n * factorial(n-1)
Jawaban 2
return factorial1(n-1, n*a)
```

Question **22**

Complete

Mark 8.00 out of
8.00

2. Jelaskan perbedaan jalan program pada **method factorial** dan **factorial1**?

Apa kesimpulan anda?.

- method factorial hanya terdapat 1 parameter sedangkan method factorial1 terdapat 2 parameter.
- method factorial mengembalikan nilai 1 sedangkan method factorial1 mengembalikan nilai a.

Comment:

Pada rekursi non-tail (Factorial), fungsi harus melakukan operasi setelah memanggil dirinya sendiri. Dalam hal ini, pemanggilan rekursif menjadi bagian terakhir dari aliran eksekusi dalam fungsi. Sebagai contoh, fungsi faktorial non-tail-recursive akan melakukan operasi perkalian n dengan hasil faktorial dari n-1 setelah memanggil dirinya sendiri.

Sementara pada rekursif tail (Factorial1), pemanggilan rekursif menjadi bagian terakhir dari aliran eksekusi dalam fungsi. Dalam hal ini, tidak ada operasi yang dilakukan setelah pemanggilan rekursif.

Tracenya Non Tail (Factorial)

nilai balik dari pemanggilan rekursif harus disimpan dalam stack, sehingga pada akhirnya stack akan terisi dengan banyak pemanggilan fungsi rekursif.

Trace Tail (Factorial1)

sehingga nilai balik dari pemanggilan rekursif dapat langsung dikembalikan ke fungsi pemanggilnya tanpa perlu menyimpannya dalam stack. Hal ini membuat pemanggilan rekursif dapat dioptimalkan menjadi iterasi, sehingga tidak ada risiko stack overflow jika pemanggilan rekursif dilakukan pada kasus yang sangat dalam.

2. Soal ini terdiri dari 3 bagian: a), b), dan c).

a). (5 poin) Method `geserKanan` berikut ini akan menggeser setiap elemen **array** `arr` ke 1 posisi di sebelah kanan, dan selanjutnya elemen di posisi terakhir akan digeser ke paling depan. Contoh: {1, 2, 3, 4} akan menjadi {4, 1, 2, 3}. Apakah isian yang tepat untuk **isianA** dan **isianB** pada method `geserKanan` berikut ini?

```
public static void geserKanan(int[] arr){
    int temp = arr[isianA];
    int i;
    for(i = isianA; i >= isianB; i--){
        arr[i] = arr[i-1];
    }
    arr[i] = temp;
}
```

Tuliskan jawaban Anda untuk **isianA** dan **isianB** di bawah ini: (Note: **JANGAN GUNAKAN SPASI PADA JAWABAN**)

isianA:



isianB:



b). (5 poin) Method `geserKanan2D` berikut ini akan menggeser setiap elemen **array of array** `arr` ke 1 posisi di sebelah kanan, dan elemen di posisi terakhir akan digeser ke paling depan. Contoh: {{1,2,3}, {4,5,6,7}, {8,9}} akan menjadi {{3,1,2}, {7,4,5,6}, {9,8}}. Asumsi method `geserKanan` pada soal a) di atas sudah diimplementasikan dengan benar dan dapat digunakan. Apakah isian yang tepat untuk **isianC** dan **isianD** pada method `geserKanan2D` berikut ini?

```
public static void geserKanan2D(int[][] arr){
    for(int i = 0; i < isianC;){
        //isianD hanya berisi 1 statement program
        isianD;
    }
}
```

Tuliskan jawaban Anda untuk **isianC** dan **isianD** di bawah ini: (Note: **JANGAN GUNAKAN SPASI PADA JAWABAN**)

isianC:



isianD:



c). (2 poin) Perhatikan implementasi method `main` berikut ini. Asumsi method `geserKanan` pada soal a) di atas sudah diimplementasikan dengan benar dan dapat digunakan. Apakah output yang akan dicetak pada potongan program di bawah ini?

```
public static void main(String[] args){
    int[] x1={1,2,3,4};
    int[] x2=x1;
    geserKanan(x1);
    geserKanan(x2);
    geserKanan(x2);
    geserKanan(x1);
    System.out.print(x2[0]);
}
```

Tuliskan output untuk program di atas di bawah ini: (Note: JANGAN GUNAKAN SPASI PADA JAWABAN)

Output:



Solusi Soal 2a :

isianA : arr.length-1

isianB : 1

isianC : arr.length

isianD : geserKanan(arr[i++])

Solusi Soal 2b :

Output : 1

[2 POIN] Perhatikan potongan kode berikut ini!

```
class Person {
    private String firstName;
    private String lastName;

    Person(String firstName, String lastName){
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public String getFirstName(){
        return this.firstName;
    }

    public String getLastName(){
        return this.lastName;
    }
}

class Queue {
    private Person[] personQueue;
    private int numPersons;
    private static final int NUM_PERSON_PER_QUEUE = 100;

    public Queue(){
        this.personQueue = new Person[NUM_PERSON_PER_QUEUE];
        this.numPersons = 0;
    }

    public void addQueue(Person p){
        this.personQueue[this.numPersons++] = p;
    }

    public void serveQueue(){
        for(int i = 0; i < this.numPersons - 1; i++){
            this.personQueue[i] = this.personQueue[i+1];
        }
        this.personQueue[--this.numPersons] = null;
    }
}
```

Hubungan antara kelas Person dengan kelas Queue disebut 

[4 POIN] Berdasarkan potongan kode pada soal sebelumnya, manakah kelas yang bersifat *mutable*? Mengapa?

class Q karena class Q terdapat kode-kode yang mengubah value dalam suatu method seperti method `addQueue()` yang menambahkan jumlah orang dalam queue.

Jawaban:

1. **[Full Mark 4 POIN]** Kelas Queue bersifat mutable karena atribut kelas ini dapat diubah setelah inisialisasi instance via method `addQueue` dan `serveQueue`.
2. **[Partial Mark 2 POIN]** Jika menjawab kedua kelas bersifat mutable dengan alasan yang benar untuk kelas Queue.
3. **[Partial Mark 1 POIN]** Jika hanya menjawab kelas Queue tanpa dilengkapi alasan atau dilengkapi dengan alasan yang kurang tepat.
4. **[0 poin]** Untuk jawaban selain 1, 2, dan 3.

Comment:

[6 POIN] Antrian pada kelas **Queue** pada soal sebelumnya bersifat **First-In-First-Out**. Tulis ulang method **addQueue()** dan **serveQueue()** pada *template* di bawah sehingga antrian tersebut menjadi bersifat **First-In-Last-Out!** (NB: Nilai penuh hanya diberikan bagi jawaban yang benar dan dapat dieksekusi pada program yang utuh).

```
public void addQueue(Person p){
    //TODO: isi jawaban anda setelah baris ini
    this.personQueue[this.numPersons++] = p;
}

public void serveQueue(){
    //TODO: isi jawaban anda setelah baris ini
    for(int i = this.numPersons-1; i >= 0; i--){
        this.personQueue[i+1] = this.personQueue[i];
    }
    this.personQueue[--this.numPersons] = null;
}
```

Jawaban:

- **[Full Mark 6 POIN]** kode ini atau sejenisnya

```
public void addQueue(Person p){
    this.personQueue[this.numPersons++] = p;
}

public void serveQueue(){
    this.personQueue[--this.numPersons] = null;
}
```

- **[Partial Mark 4 POIN]** seperti kode diatas tetapi ada typo / syntax error
- **[Partial Mark 3 POIN]** jika hanya benar addQueue atau serveQueue saja
- **[Partial Mark 2 POIN]** jika hanya benar addQueue atau serveQueue saja disertai dengan typo / syntax error
- **[0 POIN]** selain jawaban 1, 2, 3, dan 4

Comment:

serveQueue keliru

Perhatikan potongan code untuk class Exam dan Problem, serta UML diagram untuk kedua class tersebut.

```

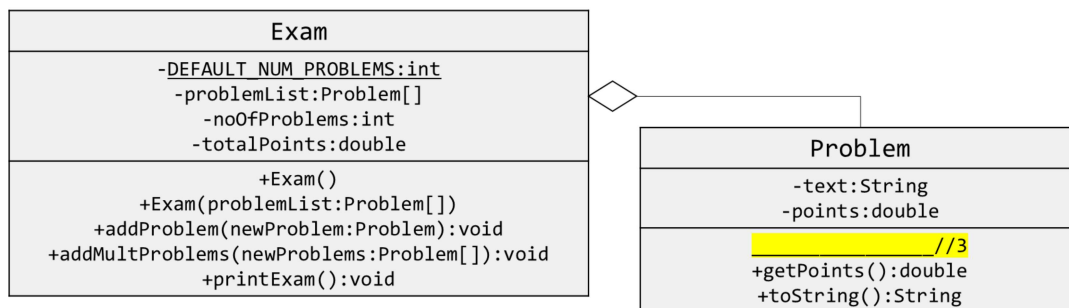
1  class Exam{
2      //1 final int DEFAULT_NUM_PROBLEMS = 5;
3      private Problem[] problemList;
4      private int noOfProblems;
5      private double totalPoints;
6
7      public Exam(){
8          this.noOfProblems = 0;
9          this.problemList = new Problem[DEFAULT_NUM_PROBLEMS];
10     }
11
12     public Exam(Problem[] problemList){
13         this.problemList = new Problem[problemList.length];
14         addMultProblems(problemList);
15     }
16
17     public void addProblem (Problem newProblem){
18         if(noOfProblems<problemList.length && newProblem!=null){
19             this.problemList[this.noOfProblems++]=newProblem;
20             this.totalPoints+=newProblem.getPoints();
21             System.out.println("New problem added.");
22         }
23         else{
24             System.out.println("Problem could not be added.");
25         }
26     }
27
28     public void addMultProblems(Problem[] newProblems){
29         for(int i=0;i<newProblems.length;i++){
30             this.addProblem(newProblems[i]);
31         }
32     }
33
34     //2{
35         System.out.printf("Exam - %d problems %.2f points\n", this.noOfProblems,
36             this.totalPoints);
37         for(int i=0;i<noOfProblems;i++){
38             System.out.printf("%d. %s\n", i+1, problemList[i]);
39         }
40     }

```

```

1  class Problem{
2      private String text;
3      private double points;
4
5      public Problem(String text, doubl
6          this.text = text;
7          this.points = points;
8      }
9
10     public double getPoints(){
11         return this.points;
12     }
13
14     public String toString(){
15         return String.format("%s (%.2
16     }
17 }

```



Lengkapilah bagian //1 dan //2 pada potongan code, dan bagian //3 pada UML diagram.

- private ☒
- public void printExam() ☒
- +Problem() ☒

Perhatikan potongan code berikut ini!

```
A Problem[] hisProblemSet = new Problem[10];
B Exam hisExam = new Exam(hisProblemSet);
C hisExam.printExam();
```

Dengan menggunakan class Exam dan Program yang sudah didefinisikan pada soal sebelumnya, jika saya mengeksekusi potongan code di atas melalui method main dari class yang berbeda, bagaimana object Exam hisExam yang diinstansiasi? Jelaskan!

*Anda dapat menjelaskan alur *code*, dan menggunakan nomor baris untuk membantu menjelaskan jawaban anda.

baris A menginisialisasikan suatu Array bernama hisProblemSet dengan object Problem yang memiliki 10 elemen.

baris B menginisialisasikan suatu object Exam bernama hisExam dengan parameter hisProblemSet sehingga memanggil method constructor Exam yang memiliki parameter yaitu di baris 12. Di dalam method Exam membuat problemList baru dengan jumlah elemen dari hisProblemSet dan pemanggilan method lain bernama addMultProblems() pada baris 14 dan mempassing value dari problemList. Di dalam method addMultProblems() pada baris 30 menambahkan problem ke array problemList dengan memanggil method addProblem() dan mempassing value dari iterasi di addMultProblems() dengan mengiterasi jika jumlah problem kurang dari jumlah elemen problemList dan newProblem tidak null, maka problem-problem dari array, menambahkan poin-poin dan print New problem added . Selain itu print Problem could not be added

baris C memanggil method printExam() dan mencetak problem-problem yang terdapat di problemList

JAWABAN YANG DIHARAPKAN:

[2 points] deskripsi object Exam yang terinisialisasi yaitu: hisExam yang terinisiasi memiliki problemSet dengan array panjang 10 tapi semuanya null dengan 0 noOfProblems dan 0 totalPoints.

[4 points] penjelasan mengenai alur code:

A. dibuat array of Problem hisProblemSet sepanjang 10 tapi setiap elemennya masih null. B. Exam hisExam diinisialisasi menggunakan constructor dengan 1 argumen dengan mengirim hisProblemSet yang memiliki length 10. Pada constructor, problemSet object Exam akan dibuat sepanjang 10 juga, lalu akan memanggil addMultProblems. addMultProblems akan mengiterasi array hisProblemSet dan memanggil addProblem dengan mengirim setiap elemen array hisProblemSet. Pada addProblem, setiap Problem yang dikirim berjenis null sehingga tidak memenuhi if block dan masuk ke else block, dan Problem tidak dapat ditambahkan. Oleh karena itu object Exam hisExam akan memiliki array hisProblemSet sepanjang 10 tapi isinya null, noOfProblems dan totalPoints 0. ******(opsional tambahan) C. Jika dicetak akan hanya mencetak isi Exam yaitu Exam - 0 problems - 0.00 points

[2/3 points] Partial score apabila menjelaskan jalan code general dengan rinci (bukan pada potongan code ini melainkan semua kondisi)

[1 points] Partial score apabila menjelaskan code A B C saja

****[-1 points]** Penalty score apabila ada statement yang salah atau tidak relevan

Comment:

anda menjelaskan code secara umum dengan 2 kondisi yang diperiksa, bukan yang terjadi pada kondisi ini. yang ditanya adlaah object Exam di pada potongan code ini. nilai parsial

Apa output dari program di bawah ini (dari menjalankan main() di FruitTest.java)? Pastikan TANPA typo atau spasi berlebih di jawaban Anda. Selain pertanyaan isian pendek, terdapat pula 1 pertanyaan B/S.

```
// Fruit.java
public class Fruit {

    public String color;
    public String taste;

    public Fruit(String color, String taste) {
        this.color = color;
        this.taste = taste;
    }

    public int f() {
        int x = 0;
        if(this instanceof Fruit) {
            x++;
        }
        if(this instanceof Object) {
            x++;
        }
        if(this instanceof Apple) {
            x++;
        }
        if(this instanceof Fuji) {
            x++;
        }
        return x;
    }

    public String toString() {
        return String.format("%s and %s fruit", taste, color);
    }
}
```

```
// Apple.java
public class Apple extends Fruit {

    public Apple() {
        super("red", "sweet");
    }

    public Apple(String color, String taste) {
        super(color, taste);
    }

    public String toString() {
        return String.format("%s and %s apple", taste, color);
    }
}

// Fuji.java
public class Fuji extends Apple {

}

// GrannySmith.java
public class GrannySmith extends Apple {
    public GrannySmith() {
        super("green", "sour");
    }
}

// GoldenDelicious.java
public class GoldenDelicious extends Apple {

    public GoldenDelicious() {
        super("yellow green", "sweet");
    }

    public String toString() {
        return "super delicious apple";
    }
}
```

```
1 // FruitTest.java
2 public class FruitTest {
3
4     public static void main(String[] args) {
5         Fruit fr = new Fruit("black", "sweet");
6         System.out.println(fr);
7         Apple ap = new Apple("yellow", "sour");
8         System.out.println(ap.f());
9         Fuji fu = new Fuji();
10        System.out.println(fu);
11        System.out.println(fu.f());
12        GoldenDelicious go = new GoldenDelicious();
13        System.out.println(go);
14    }
15
16 }
```

Output dari kode program FruitTest di atas adalah:

Line 6: ✖

Line 8: ✖

Line 10: ✖

Line 11: ✖

Line 13: ✔

Pertanyaan B/S terkait kode program di atas:



Jika method toString() pada class Fruit dihapus, program akan mengalami compilation error.

[◀ Kode Etik UTS DDP2 Sem Genap
22/23](#)

[Form Klarifikasi Nilai UTS DDP2 ▶](#)