
TEST ANNUEL – JAHRESPRÜFUNG

Informatique 1 | Informatik 1

Anweisung / Consigne :

Lesen Sie die Fragen gut durch und beantworten Sie diese **leserlich** auf den Aufgabenblättern. Für diese Prüfung dürfen Sie sind 4 Blätter (mit Vor- und Rückseite) mitnehmen, jedoch keine elektronischen Hilfen.

Tipp: Verlieren Sie bei einzelnen Fragen nicht zu viel Zeit. Beantworten Sie zuerst die Fragen, die Ihnen keine Probleme stellen, und kommen Sie später auf die für Sie schwierigeren Fragen zurück. Die Skala ist unverbindlich

Lisez attentivement la donnée et répondez de manière **lisible** aux questions. Vous avez droit pour cet examen à un aide-mémoire de 4 pages (2 feuilles recto-verso). Aucun moyen électronique n'est permis.

Un conseil : ne restez pas bloqués sur une question. Répondez tout d'abord aux questions avec lesquelles vous êtes à l'aise et revenez ensuite aux questions posant problème. Le barème indiqué est indicatif.

Question	Points	Bonus	Score
Short questions	5	0	
String understanding	4	0	
True or false	6	0	
Small program with a loop	4	0	
Working with lists	4	0	
Recursive functions	8	0	
Electric streams \neq	10	0	
MMORPG	9	0	
Total:	50	0	

This exam has 8 questions, for a total of 50 points.

Rev 1.0

Question 1 – Short questions (5 points)

Diese Frage ist in verschiedene selbständige Aufgaben unterteilt. Der Wert von jeder Aufgabe wird am Rand angegeben. Cette question est séparée en plusieurs exercices indépendants. Le nombre de points pour chaque exercice est indiqué dans la marge.

- [1 Pt] (a) Was ist eine abstrakte Klasse? Erklären Sie dies zuerst theoretisch und geben sie anschliessend ein Beispiel.
Qu'est-ce qu'une classe abstraite? Expliquez théoriquement et donnez un exemple.

.....

- [1 Pt] (b) Erklären Sie kurz was ein layout manager in Swing ist.
Expliquez brièvement à quoi sert un *layout manager* en Swing.

.....

- [1 Pt] (c) Was ist der Wert von *a* nach dem Ausführen des folgenden Codes:
Que vaut *a* après l'exécution du code suivant ?

```
1 int a = 12;
2 float b = a % 5;
3 b += (0.4 * b);
4 a = (int)(a / b);
```

(c) _____

- [1 Pt] (d) Was zeigt der folgende Code an ?
Qu'affiche le code suivant ?

```
1 for (char a = 'z'; a > 'a'; a -= 7)
2 System.out.print(" " + a + 'a');
```

.....

- [1 Pt] (e) Gegeben ist | Soit

```
1 int method(int n){
2     if (n == 1 || n == 0)
3         return n;
4     else
5         return method(n-1) + method(n-2);
6 }
```

Ist die Komplexität der oben aufgeführten Methode linear? Erklären Sie Ihren Ansatz. Est-ce que la complexité de la méthode ci-dessus est linéaire ? Expliquez votre démarche.

.....

.....

.....

.....

.....

.....

.....

Question 2 – String understanding (4 points)

Gegeben ist die folgende Funktion: | Soit la fonction suivante:

```

1 public static String mystery(String a){
2     String b = a.trim();
3     int p = b.indexOf(' ');
4
5     do{
6         String sa = b.substring(0, p);
7         System.out.println(sa);
8         b = b.substring(p+1, b.length());
9         p = b.indexOf(' ');
10
11        if(p == -1){
12            System.out.println(b.toLowerCase());
13            break;
14        }
15
16    } while(true);
17
18    return a.trim().toUpperCase();
19 }
    
```

[1 Pt] (a) *Gesucht ist der Wert für b (Zeile 2), wenn a "_the_horse_runs_fast_" entspricht.*
 Que vaut b à la ligne 2 du code ci-dessus lorsque a vaut "_the_horse_runs_fast_".

.....

[1 Pt] (b) *Was gibt diese Methode für ""Hello World ? zurück?*
 Que **retourne** cette méthode pour "Hello world" ?

.....

[2 Pt] (c) *Was zeigt diese Methode an, wenn sie "The Mammoth is GrEY" als Argument erhält?*
 Qu'affiche cette méthode si on lui donne "The Mammoth is GrEY" comme argument ?

.....

.....

.....

.....

.....

.....

Question 3 – True or false (6 points)

Es ist nicht möglich, eine endlosschleife mit einem `for` zu schreiben.
Il n'est pas possible de faire une boucle infinie avec un `for`.

True | False
 |

Eine Klasse kann zwei Interfaces implementieren.
Une classe peut implémenter 2 interfaces.

True | False
 |

Eine abstrakte Klasse kann keine Implementation von einer Methode enthalten.
Une classe abstraite ne peut pas contenir l'implémentation d'une méthode.

True | False
 |

Mehrere Klassen können von der gleichen Klasse erben.
Plusieurs classes peuvent hériter de la même classe.

True | False
 |

Ein Vektor ist viel effizienter als eine verkettete Liste um Elemente am Anfang einzufügen.
Un vecteur est bien plus efficace qu'une liste chaînée pour y insérer des éléments au début.

True | False
 |

Ein Algorithmus der Komplexität $\mathcal{O}(0)$ ist viel effizienter als ein Algorithmus $\mathcal{O}(n^2)$ für eine grosse Anzahl von Elementen.
Une algorithmme de complexité $\mathcal{O}(0)$ est plus efficace qu'un algorithmme en $\mathcal{O}(n^2)$ pour un grand nombre d'éléments.

True | False
 |

In einer `try-catch` Struktur wird der `catch` Teil ausgeführt, wenn es ein Problem mit der Kompilation im `Try`-Teil gibt.
Dans une structure `try-catch`, la partie `catch` est exécutée quand il y a eu un problème de compilation dans la partie `try`.

True | False
 |

Eine rekursive Methode ist immer wirksamer als eine iterative Methode (Schleifen).
Une méthode récursive est toujours plus efficace qu'une méthode itérative (boucle).

True | False
 |

Die Methode `Math.cos` ist statisch.
La méthode `Math.cos` est une méthode statique.

True | False
 |

Eine abstrakte Methode kann nur in einer abstrakten Klasse existieren.
Une méthode abstraite ne peut exister que dans une classe abstraite.

True | False
 |

Um von einer abstrakten Klasse zu erben, wird das Schlüsselwort `implements` benutzt.
Pour hériter d'une classe abstraite, on utilise le mot clef `implements`.

True | False
 |

Wenn die Klasse A von der Klasse B erbt und die Klasse B von der Klasse C erbt, so erbt die Klasse A nicht die Attribute welche in der Klasse C definiert sind.
Si la classe A hérite de la classe B et la classe B hérite de la classe C, A n'hérite pas des attributs définis dans C.

True | False
 |

Question 7 – Electric streams ⚡ (10 points)

Ein Stromzähler ist mit einem Computer verbunden. Er misst in regelmässigen Abständen den Stromverbrauch von verschiedenen Geräten sowie auch verschiedene Informationen wie der Strom und die Netzfrequenz. Um ein Computerprogramm zu schreiben welches mit diesen Messungen arbeitet, schlagen wir folgende Klasse vor:

Un compteur électrique est connecté à un ordinateur. Il mesure à intervalle régulier la consommation de divers appareils ainsi que différentes informations comme le courant ainsi que la fréquence de la ligne. Pour réaliser un programme informatique utilisant de telles mesures, nous proposons la classe suivante:

```

1 public class Measure {
2     double power;
3     int frequency;
4
5     public Measure(double p, int freq){
6         power = p;
7         frequency = freq;
8     }
9 }

```

[2 Pt] (a) Wie möchten folgenden Code schreiben können: | Nous aimerions pouvoir écrire le code suivant:

```

1 Measure m = new Measure(3);
2 System.out.println(m);

```

Fügen Sie nun den minimalen Code hinzu, damit Ihr Programm korrekt funktioniert und folgendes Resultat auf der Konsole anzeigt (unter der Annahme dass die Standard-Frequenz 50 ist):

Rajouter le moins de code possible pour que ce code soit valide et produise le résultat suivant sur la console (en partant du principe que par défaut la fréquence vaut 50) :

```
Measure : 3.0 [W] @ 50 [Hz]
```

.....

.....

.....

.....

.....

.....

Die Messungen sind in einer Datei nach unten aufgeführtem Format abgespeichert: Les valeurs sont stockées dans un fichier selon le format défini ci-dessous :

```

1 #,3,500
2 #,10,510
3 #,?,40,?,?,500
4 #,?,?,?,12,500

```

Die erste Zeile entspricht einer Messung von 3 Watts bei 50 Hz, die zweite Linie entspricht einer Messung von 10 Watts bei 51 Hz (beachte dass die Werte der Frequenz in der Datei mit 10 multipliziert werden). Jede Datei enthält eine variable Anzahl von Messungen und bei jedem Messintervall wird eine neue Linie in der Datei hinzugefügt. Jede Linie muss mit dem Charakter # beginnen und die verschiedenen Werte werden mit einem Komma getrennt.

Wie Sie in Zeile 3 und 4 sehen können, ist es möglich, dass die Datei das Zeichen ? enthalten kann. Dieses Zeichen entspricht nicht verfügbaren Leistungswerten welche ignoriert werden sollen. Auch in Gegenwart von unbekanntem Messungen kommt die Leistung immer vor der Frequenz und in allen Fällen kommt die Frequenz immer zuletzt.

Zur Information, die Methode `split` von `String` teilt eine Zeichenkette an der Stelle mit einem gegebenen Charakter. Das Resultat ein Array von `String`. Beispiel: `"a-lo-ha".split("-") -> {"a", "lo", "ha"}`

La première ligne correspond à une mesure de 3 watts à 50 Hz, la seconde ligne à une mesure de 10 watts à 51 Hz (notez que les valeurs en Hz sont multipliées par 10 dans le fichier). Chaque fichier contient un nombre variable de mesures et à chaque intervalle de mesure, une nouvelle ligne est ajoutée dans ce fichier. Chaque ligne doit commencer par le caractère # et les différentes valeurs sont séparées par des virgules.

Comme vous le voyez à la ligne 3 et 4, il se peut que le fichier contienne des caractères ? qui correspondent à des valeurs de puissance indisponibles qui doivent être ignorées. Même en présence de mesures inconnues, la puissance vient toujours avant la fréquence et, dans tous les cas, la fréquence apparaît toujours en dernier.

Pour rappel, la méthode `split` de la classe `String` permet de découper un `String` à l'aide d'un caractère donné. Le type de résultat est un tableau de `String`. Exemple: `"a-lo-ha".split("-") -> {"a", "lo", "ha"}`

- [2 Pt] (a) *Es ist möglich, das Linien mit Messungen nicht gültig sind. Die zwei möglichen Ursachen sind, dass die Zeile nicht mit # beginnen oder dass die Zeile nur unbekannte Werte, in Folge eines Problems mit dem Sensor, enthält. Beachte, das angenommen wird, dass wenn ein numerischer Wert in der Linie enthalten ist, muss der andere Wert auch da sein. Implementieren die Funktion `isValid`, die als Parameter eine Linie der Datei mit den Messungen erhält und `true` zurückgibt, wenn die Linie gültig ist.*

Il se peut que des lignes de mesures ne soient pas valides. Les deux problèmes possibles sont que la ligne ne commence pas par # ou que la ligne ne contient que des valeurs inconnues en raison d'un problème de capteurs. Notez que l'on fait l'hypothèse que si une valeur numérique est présente dans la ligne, la seconde le sera également. Implémentez la fonction `isValid` qui prend comme paramètre une ligne du fichier de mesure et retourne `true` lorsque celle-ci est valide.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

- [3 Pt] (b) *Implementieren Sie nun eine Funktion `decodeMeasure`, welche den Inhalt einer kompletten Messung, durch dekodieren der Linie zurückgibt. Die Linie wird als gültig angenommen. Der Rückgabewert ist vom Typ `Measure`.* Implémentez maintenant une fonction `decodeMeasure` qui doit retourner le contenu d'une mesure complète en décodant une ligne que l'on supposera valide. La valeur retournée est de type `Measure`.

défaut : 0 pour l'expérience, 10 pour les points de vie et 5 pour l'armure. Cette classe, qui ne doit pas être instanciable, doit aussi comporter un constructeur qui permet de spécifier le nom de l'unité.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

- [3 Pt] (b) Schreiben Sie nun eine Klasse *MovingUnit* welche von der Klasse *Unit* erbt. Diese Klasse, welche auch nicht instanziiert werden kann, besitzt ein zusätzliches Integer Attribut namens *MP*, welches die Zahl der Bewegungspunkte darstellt. Ausserdem muss diese Klasse sicherstellen, dass alle Klassen, welche von dieser Klasse erben, eine Methode *move* implementieren. Die Methode *move* erhält zwei Ganzzahlen als Parameter.

Écrivez maintenant une classe *MovingUnit* qui hérite de la classe *Unit*. Cette classe, qui elle aussi ne doit pas être instanciable, possède un attribut entier supplémentaire nommé *MP* pour représenter le nombre de points de mouvement. De plus, cette classe doit faire en sorte que toutes les classes qui hériteront de celle-ci seront obligées d'avoir une procédure *move* qui prend deux entiers en paramètres.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

- [1 Pt] (c) Das unten aufgeführte UML Diagramm repräsentiert einen Teil der Hierarchie des Spiels. Dies ermöglicht es Ihnen, einen besseren Überblick über die Vererbung der Klassen zu erhalten. Es wird darauf hingewiesen, dass die Beschreibung der Klassen **unvollständig** ist.

Man möchte nun Spezifikationen für zwei Kategorien von Einheiten hinzufügen. Die erste sind die Pfleger (*Healer*) welche eine Methode namens *heal* besitzen müssen. Die Methode *heal* erhält als Parameter eine Einheit. Beispiele dieser Klasse im oben aufgeführten UML-Diagramm sind *Hospital*, *Priest* und *Paladin*. Die zweiten welche spezifiziert werden sind die Krieger (*Warrior*) welche eine Methode namens *sword* besitzen müssen. Die Methode *sword* erhält auch eine Einheit als Parameter (zum Beispiel *Paladin* oder *Fencer*). So gibt es einige Klassen welche *Healer* oder *Warrior* oder beides sind. Diese Klassen können sowohl in der Kategorie *MovingUnit* oder *StaticUnit* sein. Erklären Sie in einem Satz welche Lösung benutzt werden könnte, um diese Spezifikation zu realisieren.

