
TEST INTERMÉDIAIRE – ZWISCHENPRÜFUNG

SOLUTION

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 1 Blatt (mit Vorseite) 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 1 pages (1 feuille recto). 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 | 9 | 0 | |
| Loops analysis | 6 | 0 | |
| Code comprehension | 5 | 0 | |
| Function prototypes | 4 | 0 | |
| Writing some functions | 10 | 0 | |
| Functions working with text | 10 | 0 | |
| Counting days | 11 | 0 | |
| Total: | 55 | 0 | |

This exam has 7 questions, for a total of 55 points.

Rev 1.0ρ

Question 1 – Short questions (9 points)

Diese Frage ist in verschiedene selbständige Aufgaben unterteilt. Die Anzahl Punkte von jeder Aufgabe wird am Rand vermerkt. Cette question est séparée en plusieurs exercices indépendants. Le nombre de point pour chaque exercice est indiqué dans la marge.

[2 Pt] (a) Gegeben sind die folgenden Deklarationen: | Soient les déclarations suivantes:

```
int a = 'd';
short b = (byte) 97;
char c = (char) 4;
double d = 2.5;
String e = "bar";
```

Geben Sie an, ob die nachstehenden Ausdrücke korrekt sind oder nicht. Wenn sie korrekt sind, bestimmen Sie den **Typ** und den **Wert**. Zur Erinnerung, der ASCII-Wert von 'a' ist 97.

Indiquez si les expressions suivantes sont correctes. Si elles sont correctes, donnez le **type** et la **valeur** des expressions suivantes. Pour rappel, la valeur ASCII de 'a' est 97.

(a) $d + b$

(a) double, 99.5

(b) $e + (\text{char})(b + c)$

(b) String, "bare"

(c) $a < 3 \ \&\& \ ((0 \times \text{BEEF} \gg 32) \neq 2)$

(c) boolean, false

(d) $!(c > 'c') ? e + a : e$

(d) String, "bar100"

[3 Pt] (e) Wahr oder falsch? | Vrai ou faux ?

Numerische Datentypen (int, byte, ...) haben in Java immer Vorzeichen.
Les types numériques (int, byte, ...) sont toujours signés en Java

| True | False |
|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Es ist möglich, eine Variable ohne Type zu deklarieren.
Il est possible de déclarer une variable sans type.

| True | False |
|--------------------------|-------------------------------------|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Der Compiler entdeckt Typisierungsfehler.
Le compilateur détecte les erreurs de typage

| True | False |
|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Eine Funktion kann mehr als einen Rückgabewert zurückgeben.
Il est possible de déclarer des variables à peu près n'importe où dans le corps d'une fonction.

| True | False |
|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Die Operatoren *, / % und + haben dieselbe Vorrangsstufe (Priorität).
Les opérateurs *, / % et + ont le même niveau de précedence (priorité)

| True | False |
|--------------------------|-------------------------------------|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Im Ausdruck $a ? b : c$, müssen die Variablen b und c von Typ String sein.
Dans l'expression $a ? b : c$, les variables b et c peuvent être de type String

| True | False |
|-------------------------------------|--------------------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> |

[2 Pt] (f) *Gesucht ist der Wert der folgende Ausdrücke im Hexadezimalsystem.*
 Donnez la valeur des expressions suivantes **en hexadécimal**.

1) $(0b1011 \ll 16) + (0b0010 * 2)$

1) 0xB0004

2) $0xF0F0 \& 0xBEBE$

2) 0xB0B0

[2 Pt] (g) *Schreiben Sie das Resultat im Dezimalsystem.*
 Donnez le résultat des opérations suivantes en décimal.

1) $(4 / 3 + 4 \% 2) \% 2$

1) 1

2) (byte) $(4 * 60 + 272) / 3 + 2$

2) 2

Question 2 – Loops analysis (6 points)

Welche Resultate werden auf der Konsole angezeigt?

Que vont afficher **exactement** les boucles suivantes sur la console ?

(a)

```
1 char k = 'b';
2 for(int i = 10; i >= 5; i -= 2){
3     System.out.print(k);
4 }
```

Lösung | Solution:

Solution:
bbb

(b)

```
1 int i = 0;
2
3 do{
4     i += 4;
5
6     if (i / 10 == 1)
7         break;
8
9     System.out.print(i - 1 + ",");
10 }while (true);
11
12 System.out.println(i - 1);
```

Lösung | Solution:

Solution:
3,7,11

(c)

```
1 int n = 50;
2 int i = 1;
3
4 for (i = i - 1; n >= 15; i += 12) {
5     n -= i;
6     System.out.println(i + ", " + n);
7 }
```

Lösung | Solution:

Solution:
0, 50
12, 38
24, 14

Question 3 – Code comprehension (5 points)

[2 Pt] (a) Gegeben sei folgende Funktion: | Soit la fonction suivante :

```

1  public static String foo(int a) {
2      String result = "";
3      int i = 0;
4
5      while(i < a) {
6          switch(i) {
7              case 0:
8                  result += 'g';
9                  i++;
10                 i++;
11                 break;
12                case 1:
13                    result += 'p';
14                    i--;
15                    break;
16                case 2:
17                    result += (char)('n' + 1);
18                    result += (char)('n' + 1);
19                    i = -5;
20                    break;
21                default:
22                    result += (char) 100;
23                    i = a;
24                    break;
25            }
26        }
27
28        return result;
29    }

```

Was gibt diese Funktion zurück, wenn `foo(21)` (ASCII-Tabelle hierunter) aufgerufen wird? Que retourne cette fonction lors de l'appel `foo(21)` (vous pouvez utiliser la table ASCII ci-dessous)?

| | | | | | | | | | | | | | | | |
|-----|---------|-----|---------|-----|----|-----|---|-----|---|-----|---|-----|---|-----|---|
| 00h | ␣ (nul) | 10h | ▶ (dle) | 20h | □ | 30h | 0 | 40h | @ | 50h | P | 60h | ‘ | 70h | p |
| 01h | Ⓢ (soh) | 11h | ◀ (dc1) | 21h | ! | 31h | 1 | 41h | A | 51h | Q | 61h | a | 71h | q |
| 02h | Ⓢ (stx) | 12h | ‡ (dc2) | 22h | " | 32h | 2 | 42h | B | 52h | R | 62h | b | 72h | r |
| 03h | ▼ (etx) | 13h | ‡ (dc3) | 23h | # | 33h | 3 | 43h | C | 53h | S | 63h | c | 73h | s |
| 04h | ◆ (eot) | 14h | ‡ (dc4) | 24h | \$ | 34h | 4 | 44h | D | 54h | T | 64h | d | 74h | t |
| 05h | ♣ (enq) | 15h | § (nak) | 25h | % | 35h | 5 | 45h | E | 55h | U | 65h | e | 75h | u |
| 06h | ♠ (ack) | 16h | – (syn) | 26h | & | 36h | 6 | 46h | F | 56h | V | 66h | f | 76h | v |
| 07h | • (bel) | 17h | ‡ (etb) | 27h | ' | 37h | 7 | 47h | G | 57h | W | 67h | g | 77h | w |
| 08h | ▣ (bs) | 18h | ↑ (can) | 28h | (| 38h | 8 | 48h | H | 58h | X | 68h | h | 78h | x |
| 09h | ␣ (tab) | 19h | ↓ (em) | 29h |) | 39h | 9 | 49h | I | 59h | Y | 69h | i | 79h | y |
| 0Ah | ▣ (lf) | 1Ah | ␣ (eof) | 2Ah | * | 3Ah | : | 4Ah | J | 5Ah | Z | 6Ah | j | 7Ah | z |
| 0Bh | ♂ (vt) | 1Bh | ← (esc) | 2Bh | + | 3Bh | ; | 4Bh | K | 5Bh | [| 6Bh | k | 7Bh | { |
| 0Ch | ␣ (np) | 1Ch | Ⓛ (fs) | 2Ch | , | 3Ch | < | 4Ch | L | 5Ch | \ | 6Ch | l | 7Ch | |
| 0Dh | Ⓜ (cr) | 1Dh | ↔ (gs) | 2Dh | - | 3Dh | = | 4Dh | M | 5Dh |] | 6Dh | m | 7Dh | } |
| 0Eh | Ⓜ (so) | 1Eh | ▲ (rs) | 2Eh | . | 3Eh | > | 4Eh | N | 5Eh | ^ | 6Eh | n | 7Eh | ~ |
| 0Fh | Ⓜ (si) | 1Fh | ▼ (us) | 2Fh | / | 3Fh | ? | 4Fh | O | 5Fh | _ | 6Fh | o | 7Fh | △ |

Figure 1 – The beginning of the ASCII table (in hexadecimal)

Solution: "good"

[3 Pt] (b) *Gegeben sei* | Soit la fonction ci-dessous:

```

1  public static String understanding1() {
2      String s = "";
3      for(char i = 'a'; i <= 'z'; i++) {
4          for(int j = 1; j <= 5; j++) {
5              for(char k = 'a'; k <= 'z'; k++) {
6                  s += i + "-" + j+ "-" + k + '\n';
7              }
8          }
9      }
10     return s;
11 }

```

Tip: das Zeichen \n macht einen Zeilenumbruch auf der Konsole, wenn es sich in einem System.out.println angezeigten String befindet. Geben Sie die Werte der ersten drei und der letzten drei angezeigten Zeilen an, wenn der Ausdruck System.out.println(understanding1()); ausgeführt wird.

Pour mémoire, le caractère \n fait un retour à la ligne sur la console lorsqu'il se trouve dans un String affiché par un System.out.println. Donnez les valeurs des trois premières et trois dernières lignes affichées lorsque l'on exécute l'expression System.out.println(understanding1());

Solution:

```

a-1-a
a-1-b
a-1-c
...
z-5-x
z-5-y
z-5-z

```

Question 4 – Function prototypes (4 points)

[2 Pt] (a) *Erklären Sie, warum die folgenden Aufrufe von foo möglich sind, obwohl die Parameter verschieden sind?*
 Pourquoi est-il possible de faire les appels suivants de foo alors que les paramètres sont différents ?

```

1  int i;
2
3  i = foo(1, 2.0f, 3);
4  i = foo("bar", 0xBEEF);
5  i = foo('3', 2, true);
6  i = foo();

```

Solution: Because there must be several prototypes for the function named foo with different number and/or types of parameters..

[2 Pt] (b) *Geben Sie den Prototyp der Funktion(en) foo, wenn es keinen automatischen cast gibt.*
 Donnez le prototype de la (ou des) fonction(s) foo si l'on considère qu'il n'y a pas de cast automatique.

Solution:

- public static int foo(int a, float b, int c)
- public static int foo(String a, int b)
- public static int foo(char a, int b, boolean c)
- public static int foo()

Question 5 – Writing some functions (10 points)

- [2 Pt] (a) Schreiben Sie eine Funktion, die drei Ganzzahlen als Parameter erhält und `true` zurückgibt, falls mindestens eines der Argumente positiv ist.

Écrivez une fonction qui prend trois entiers en paramètre et qui retourne `true` si au moins l'un des trois arguments est positif.

Solution:

```

1 public static boolean positive(int a, int b, int c){
2     if(a > 0 || b > 0 || c > 0)
3         return true;
4     return false;
5 }
```

- [3 Pt] (b) Schreiben sie eine Funktion, die eine Zahl zwischen 0 und 15 (mit dem kleinstmöglichen Typ) als Parameter erhält und das entsprechende Hexadezimalzeichen zurückgibt oder 'X', falls die erhaltene Zahl nicht zwischen 0 und 15 liegt.

Écrivez une fonction qui prend en paramètre un nombre entre 0 et 15 (avec le type le plus petit possible) et retourne le caractère hexadécimal correspondant, ou 'X' si le nombre reçu n'est pas entre 0 et 15.

Solution:

```

1 public static char number2hex(byte n){
2     if (n >= 0 && n <= 9) {
3         return (char) ('0' + n);
4     }
5
6     if (n >= 10 && n <= 15) {
7         return (char) ('a' + n);
8     }
9     return 'X';
10 }
```

- [5 Pt] (c) Im Tischtennis beginnt ein Match mit dem Spielstand 0-0. Das Spiel ist beendet, wenn einer der beiden Spieler 11 Punkte erreicht und mit mindestens zwei Punkten Vorsprung führt (z. B. 3-11: der zweite Spieler hat gewonnen). Bei einem Spielstand von 10-10 wird so lange weitergespielt, bis einer der Spieler mit mind. zwei Punkten Vorsprung führt, d. h. frühestens beim Spielstand 12-10.

Schreiben Sie eine Funktion, die zwei Zahlen als Argument erhält und 0 zurückgibt, falls der Match noch nicht beendet ist, 1 falls der Spieler 1 gewonnen hat und 2 falls der Spieler 2 gewonnen hat. Wenn der Spielstand ungültig ist, gibt die Funktion -1 zurück.

Au tennis de table, une manche commence avec le score 0-0. La manche est terminée lorsque l'un des deux joueurs obtient 11 points avec au moins 2 points d'écart (par exemple 3-11, le second joueur a gagné). En cas de score 10 à 10, le jeu se poursuivra jusqu'à ce que l'un des joueurs obtienne deux points de plus que l'autre, à savoir au plus tôt avec un score de 12 à 10.

Écrivez une fonction qui reçoit deux nombres en argument et qui retourne 0 si la manche est encore en cours, 1 si le joueur 1 a gagné et 2 si le second joueur a gagné. Si le score n'est pas valide, la fonction retournera -1.

Solution:

```
1  public static int tableTennis(int a, int b) {
2      // Illegal
3      if (a < 0 || b < 0)
4          return -1;
5
6      // Still playing
7      if (a < 10 && b < 10)
8          return 0;
9
10     // First player wins, simple case
11     if (a == 11 && b <= 9)
12         return 1;
13
14     // Second player wins, simple case
15     if (b == 11 && a <= 9)
16         return 2;
17
18     // Still playing, but after limit
19     if (Math.abs(a - b) < 2)
20         return 0;
21     // Should never happen because impossible, trickier to find this one
22     else if (Math.abs(a - b) > 2)
23         return -1;
24
25     if (a > b)
26         return 1;
27     else
28         return 2;
29 }
```

Question 6 – Functions working with text (10 points)

- [4 Pt] (a) Gegeben ist die Funktion `letterCount(String s, char c)`, die zurückgibt, wie oft der Buchstabe `c` in der Zeichenkette `s` vorkommt. Schreiben Sie ausgehend von dieser Funktion eine Funktion, die die Anzahl der Vorkommnisse des häufigsten Buchstaben zurückgibt. Beispiel: `letterCountMax("alibaba")` gibt 3 zurück.
- Soit la fonction `letterCount(String s, char c)` qui retourne le nombre d'occurrences de la lettre `c` dans la chaîne `s`. A l'aide de cette fonction, écrivez une fonction qui retourne le nombre d'occurrences de la lettre la plus représentée. Par exemple, `letterCountMax("alibaba")` retourne 3.

Solution:

```

1  public static int letterCountMax(String s){
2      int max = 0;
3      for (char c = 'a'; c <= 'z'; c++)
4      {
5          int tmp = letterCount(s, c);
6          if (tmp > max)
7          {
8              max = tmp;
9          }
10     }
11
12     return max;
13 }

```

- [6 Pt] (b) Nehmen Sie an, dass diese Funktionen in Ihrem Code zur Verfügung stehen. | Considérez que vous avez à disposition dans votre code les fonctions suivantes:

```

1  /** Returns the length (number of letters) of the String s. */
2  public static int stringLength(String s);
3
4  /** Returns the char at position pos in s. */
5  public static char stringCharAt(String s, int pos);

```

Schreiben Sie eine Funktion, mit der überprüft werden kann, ob ein `String` komplex genug ist, um als Passwort verwendet werden zu können. Diese Funktion muss `true` zurückgeben, falls das Passwort die erforderliche Komplexität aufweist, und `false`, falls nicht. Die Komplexität wird als ausreichend betrachtet, wenn das Passwort aus mindestens 10 Zeichen der ASCII-Tabelle besteht (zwischen dem Leerzeichen und dem Symbol `~` eingeschlossen), davon mindestens zwei Zahlen.

Écrivez une fonction permettant de vérifier si un `String` possède la complexité requise pour être un mot de passe. Cette fonction doit retourner `true` si le mot de passe possède la complexité requise et `false` autrement. La complexité est considérée comme suffisante si le mot de passe possède au moins 10 caractères de la table ASCII (compris entre l'espace qui est le caractère à la position 32 et le symbole `~`, y.c.), dont au moins deux chiffres.

Solution:


```
1 public static boolean passComplexity(String s){
2     int nNumbers = 0;
3     int nChars = stringLength(s);
4
5     if(nChars < 10)
6         return false;
7
8     for (int i = 0; i < nChars ; i++) {
9         char c = charAt(s, i);
10        if(c >= '0' && c <= '9')
11            nNumbers++;
12
13        if(c < 32 || c > 126)
14            return false;
15    }
16
17    return nNumbers >= 2;
18 }
```

△ Turn page →

Question 7 – Counting days (11 points)

Man möchte eine Applikation entwickeln, welche die Anzahl an Tagen zwischen zwei Kalenderdaten zählt, zum Beispiel um den möglichen Geburtstermin zu ermitteln. Dazu verlangt man von Ihnen, zwei Funktionen zu schreiben.

On désire réaliser une application permettant de compter combien de jours séparent deux dates dans le calendrier, par exemple pour déterminer la date probable d'un accouchement. Pour ce faire, on vous demande d'écrire deux fonctions.

- [3 Pt] (a) Die erste Funktion namens `ndays` erlaubt es, die Anzahl Tage eines Monats zu erhalten. Sie nimmt also einen Monat (`int`) als Argument und gibt dessen Anzahl Tage zurück. Dazu ist es interessant zu wissen, dass bis und mit Juli alle ungeraden Monate 31 Tage haben und die geraden Monate 30 Tage (ausgenommen Februar, der für diese Übung immer 28 Tage hat). Ab August, haben die geraden Monate 31 Tage und die ungeraden 30 Tage. Achtung, es ist nicht erlaubt ein `switch` oder eine `if` Struktur mit allen Monaten zu benutzen.

La première fonction, nommée `ndays`, permettra de déterminer combien de jours il y a pour n'importe quel mois de l'année. Elle prend donc un mois (`int`) en argument et retourne le nombre de jours pour mois donné. Pour implémenter cette fonction, il faut savoir que jusqu'au mois de juillet y compris, les mois impairs ont 31 jours alors que les mois pairs ont 30 jours (sauf février qui, pour l'exercice, a toujours 28 jours). À partir d'août, les mois pairs ont 31 jours alors que les mois impairs 30 jours. Attention, vous ne pouvez pas utiliser un `switch` ou une structure de `if` et énumérer tous les mois de l'année!

Solution:

```
1  static public int ndays(int month){
2      if(month == 2){
3          return 28;
4      }
5
6      if(month <= 7)
7      {
8          if(month % 2 == 0)
9              return 30;
10         else
11             return 31;
12     }
13
14     if(month % 2 == 0)
15         return 31;
16     else
17         return 30;
18 }
```

- [5 Pt] (b) Die zweite Funktion heisst `getdays` und nimmt als Parameter zwei Daten (wobei jedes aus einem Tag und einem Monat besteht) und gibt die Anzahl Tage zwischen den beiden Daten zurück. Für diese Übung ist das zweite Datum immer später als das erste. Einige Funktionsaufrufe sind unten als Beispiel gegeben:

La seconde fonction, nommée `getdays`, prend comme argument deux dates (composée chacune d'un jour et d'un mois), et retourne le nombre de jours entre les deux. Pour simplifier l'exercice, partez du principe que la seconde date est toujours postérieure à la première lors de l'appel de la fonction. Quelques exemples d'appel de cette fonction sont donnés ci-après:

```

1  getdays(1,1,1,1) // → 0
2  getdays(1,1,2,1) // → 1
3  getdays(1,1,20,1) // → 19
4  getdays(1,1,1,2) // → 31
5  getdays(1,1,2,2) // → 32
6  getdays(1,1,31,3) // → 30+28+31
7  getdays(1,1,31,12) // → 364

```

Solution:

```

1  static public int getdays(int sDay, int sMonth, int eDay, int eMonth) {
2      int mDiff = eMonth - sMonth;
3      int totalDays = 0;
4
5      if(sMonth == eMonth)
6          return eDay - sDay;
7
8      // Number of days during complete months
9      for (int i = sMonth + 1; i < eMonth; i++) {
10         totalDays += ndays(i);
11     }
12
13     // Complete days in the starting month
14     int daysStart = ndays(sMonth) - sDay;
15
16     // Complete days in the last month
17     int daysEnd = eDay;
18
19     totalDays += daysEnd + daysStart;
20
21     return totalDays;
22 }

```

- [2 Pt] (c) *Erklären Sie - ohne zu implementieren - wie Sie mit vorheriger Funktion das Datum eines Ereignisses, das 280 Tage nach einem gegebenen Datum passiert, bestimmen können (angenommen das gegebene Datum ist vor dem 26. März).*

Expliquez – sans l’implémenter – comment vous pourriez, à l’aide de la fonction ci-dessus, déterminer la date d’un événement se passant 280 jours après une date donnée (si l’on part du principe que la date de départ se situe avant le 26 mars).

Solution: One can simply call the function repeatedly and increase the second date until we get the correct amount of days. During the call, the first argument of the function is the event date and the second date being just one day after the event date at first. Then, in a loop, we increase the second date until the number returned is 280.

- [1 Pt] (d) *Warum ist dies schwieriger zu Implementieren, wenn das gegebene Datum nach dem 26. März ist? Pourquoi est-ce plus difficile à implémenter si la date peut se situer après le 26 mars ?*

Solution: Because we have to consider cases where the second date is before the first one (when we cross the border of the year).

Fin|Ende
