Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Prüfung

Informatik I (D-BAUG)

Felix Friedrich, Hermann Lehner

ETH Zürich, 29.1.2018.

Name, Vorname:	. . . . .
Legi-Nummer:	. . . . .

Ich bestätige mit meiner Unterschrift, dass ich diese Prüfung unter regulären Bedingungen ablegen konnte, und dass ich die allgemeinen Richtlinien gelesen und verstanden habe.

I confirm with my signature that I was able to take this exam under regular conditions and that I have read and understood the general guidelines.

Unterschrift:

Allgemeine Richtlinien:

1. Dauer der Prüfung: 60 Minuten.
2. Erlaubte Unterlagen: Wörterbuch (für gesprochene Sprachen). 4 A4 Seiten handgeschrieben oder $\geq$ 11pt Schriftgrösse.
3. Benützen Sie einen Kugelschreiber (blau oder schwarz) und keinen Bleistift. Bitte schreiben Sie leserlich. Nur lesbare Resultate werden bewertet.
4. Lösungen sind direkt auf das Aufgabenblatt in die dafür vorgesehenen Boxen zu schreiben (und direkt darunter, falls mehr Platz benötigt wird). Ungültige Lösungen sind deutlich durchzustreichen! Korrekturen bei Multiple-Choice Aufgaben bitte unmissverständlich anbringen!
5. Es gibt keine Negativpunkte für falsche Antworten.
6. Störungen durch irgendjemanden oder irgendetwas melden Sie bitte sofort der Aufsichtsperson.
7. Wir sammeln die Prüfung zum Schluss ein. Wichtig: stellen Sie unbedingt selbst sicher, dass Ihre Prüfung von einem Assistenten eingezogen wird. Stecken Sie keine Prüfung ein und lassen Sie Ihre Prüfung nicht einfach am Platz liegen. Dasselbe gilt, wenn Sie früher abgeben wollen: bitte melden Sie sich lautlos, und wir holen die Prüfung ab. Vorzeitige Abgaben sind nur bis 15 Minuten vor Prüfungsende möglich.
8. Wenn Sie zur Toilette müssen, melden Sie dies einer Aufsichtsperson durch Handzeichen.
9. Wir beantworten keine inhaltlichen Fragen während der Prüfung. Kommentare zur Aufgabe schreiben Sie bitte auf das Aufgabenblatt.

General guidelines:

1. *Exam duration: 60 minutes.*
2. *Permitted examination aids: dictionary (for spoken languages). 4 A4 pages hand written or $\geq$ 11pt font size*
3. *Use a pen (black or blue), not a pencil. Please write legibly. We will only consider solutions that we can read.*
4. *Solutions must be written directly onto the exam sheets in the provided boxes (and directly below, if more space is needed). Invalid solutions need to be crossed out clearly. Provide corrections to answers of multiple choice questions without any ambiguity!*
5. *There are no negative points for false answers.*
6. *If you feel disturbed by anyone or anything, let the supervisor of the exam know immediately.*
7. *We collect the exams at the end. Important: you must ensure that your exam has been collected by an assistant. Do not take any exam with you and do not leave your exam behind on your desk. The same applies when you want to finish early: please contact us silently and we will collect the exam. Handing in your exam ahead of time is only possible until 15 minutes before the exam ends.*
8. *If you need to go to the toilet, raise your hand and wait for a supervisor.*
9. *We will not answer any content-related questions during the exam. Please write comments referring to the tasks on the exam sheets.*

Question:	1	2	3	4	5	6	Total
Points:	14	10	12	10	10	4	60
Score:							

Aufgabe 1: Java Grundlagen (14P)

Die folgenden Fragen beziehen sich auf den Code oben auf der gegenüberliegenden Seite.

The following questions refer to the code on top of the opposite side.

/1P (a) Kreisen Sie direkt im Code die Stellen ein, an denen der Variable `sum` ein Wert zugewiesen wird.

Encircle directly in the code the position, where a value is being assigned to the variable `sum`.

/2P (b) Betrachten Sie die Variable `carry`. (i) Welche Werte kann diese Variable über die gesammte Laufzeit des Programms annehmen? (ii) Geben Sie den kleinstmöglichen Typ für diese Variable an, so dass sie diese Werte enthalten kann (statt `int`).

Examine the variable `carry`. *(i) What are possible values of this variable over the entire runtime of the program? (ii) State the smallest-possible type for this variable to hold those values (instead of* `int`*).*

(i): (ii):

/2P (c) Was ist der Wert von Variable `i` wenn die Programmausführung bei Zeile 19 angelangt ist?

What is the value of variable `i` *when the program execution reaches line 19?*

◯ 1 ◯ 0 ◯ -1 ◯ Keiner (ungültig)/*None (invalid)*

/3P (d) Angenommen, die Array-wertigen Variablen `x` und `y` sind belegt wie folgt. Welchen Wert hat Variable `r` dann nach der Zuweisung `r = m(x, y)`? Achten Sie auf die richtige Anzahl Array-Elemente für Variable `r`. Ignorieren Sie die fehlende Zeile 18: betrachten Sie sie als leer.

Assume the array valued variables `x` *and* `y` *are assigned like in the following. What is then the value of* `r` *after the assignment* `r = m(x, y)`*? Make sure to provide the correct number of array elements for variable* `r`*. Ignore the missing line 18: consider it empty.*

x =	3	5
	0	1

y =	4	7
	0	1

r =

/2P (e) Der Aufruf von `r = m(x, y)` führt zu einem Absturz in Zeile 9 für folgende Belegung der Variablen `x` und `y`. Schreiben Sie in Zeile 2 im Code eine zusätzliche Vorbedingung, welche für die Parameter gelten muss.

The call of `r = m(x, y)` *results in a crash on line 9, given the following assignments to variables* `x` *and* `y`*. Write in line 2 in the code another precondition that needs to hold for the parameters.*

x =	3	5	8
	0	1	2

y =	4	7
	0	1

```
1   // PRE: Each element of a and b is a number between 0 and 9
2   // PRE:
3   static int[] m(int[] a, int[] b){
4       int size = a.length;
5       // note: all elements of an int array are initialized with zero
6       int[] res = new int[size + 1];
7       int carry = 0;
8       for (int i = size - 1; i >= 0; --i){
9           int sum = a[i] + b[i] + carry;
10          if (sum >= 10) {
11              sum = sum - 10;
12              carry = 1;
13          } else {
14              carry = 0;
15          }
16          res[i + 1] = sum;
17      }
18
19      return res;
20  }
```

(f) Beschreiben Sie in einem Satz, welche spezifische Aufgabe von der Methode `m` gelöst wird.

Explain in one sencence, what specific task is solved by method `m`. /2P

(g) Der Aufruf von `r = m(x, y)` führt zum falschen Ergebnis für die folgende Belegung der Variablen `x` und `y`. Im Code auf Zeile 18 fehlt eine Anweisung. Ergänzen Sie diese direkt im Code.

The call of `r = m(x, y)` *leads to a wrong result, given the following assignments to the variables* `x` *and* `y`. *In the code on line 18, a statement is missing. Fill it in directly in the code.* /2P

x = | 8 | 5 |
0 1

y = | 4 | 7 |
0 1

Aufgabe 2: Methoden und Rekursion (10P)

Im folgenden ist das **Pascalsche Dreieck** abgebildet. Jede Zeile $r > 0$ enthält $r+1$ Zahlen welche durch Addition von Zahlen aus Zeile $r-1$ gebildet werden. Die erste Zeile $r = 0$ enthält die Zahl 1.
Die Zahl $p(r, c)$ in Zeile $r > 0$ und Spalte $0 \leq c \leq r$ berechnet sich als die Summe der beiden Zahlen in Zeile $r-1$ und Spalten c und $c-1$, also den zwei Zahlen unmittelbar darüber und links davon, siehe nachfolgende Abbildung.
Damit dies auch am linken und rechten Rand funktioniert, werden die Zeilen jeweils gedanklich um eine Null links und rechts ergänzt. Wir definieren also $p(r, c) = 0$ für alle $c < 0$ und für alle $c > r$.

Below **Pascal's Triangle** *is shown. Each row $r > 0$ contains $k+1$ numbers that are formed by adding numbers from row $k-1$. The first row $r = 0$ contains the number 1.*
The number $p(r, c)$ in row $r > 0$ and column $0 \leq c \leq r$ is computed as the sum of the two numbers in row $r-1$ an columns c and $c-1$, i.e. the sum of the numbers immediately above and left from it.
In order to make this work at the left and right borders, the rows are complemented by zeros to the left and to the right. We thus define $p(r, c) = 0$ for all $c < 0$ and for all $c > r$.

Spalten / columns →

Zeilen / rows ↓ Index	0	1	2	3	4
0	1				
1	1	1			
2	1	2	1		
3	1	3	3	1	
4	1	4	6	4	1

(a) Ergänzen Sie die folgende Methode so, dass sie die Zahl $p(r, c)$ in Zeile r und Spalte c berechnet. Zeilen und Spalten werden von 0 nummeriert.

Complement the following method such that it computes the number $p(r, c)$ in row r and column c. Rows and columns are numerated from 0.

/6P

```
// Compute number in Pascal's triangle at row r>=0 and column c
public static int p(int r, int c){
    [                                        ]{
        return 1;
    }
    [                                        ]{
        return 0;
    }
    [                                        ]
}
```

(b) Geben Sie einen kurzen informellen Beweis dafür an, dass Ihre Methode für alle Eingaben (Parameter) $r \geq 0$, $c \geq 0$ terminiert.

Provide a short informal proof that your method will terminate for all inputs (parameters) $r \geq 0$, $c \geq 0$.

/4P

/12P

Aufgabe 3: Dynamische Datenstrukturen (12P)

Eine Multi-Menge ist eine Menge, bei der jedes Element mehrfach vorkommen kann. Die Reihenfolge der Elemente in einer Menge ist irrelevant.
In der dargestellten Implementation wird diese als verkettete Liste gespeichert, bei der jedes Element zusätzlich die Häufigkeit seines Auftretens in der Menge speichert. Nachfolgend ist die Datenstruktur eines einzelnen Elementes der Menge dargestellt.

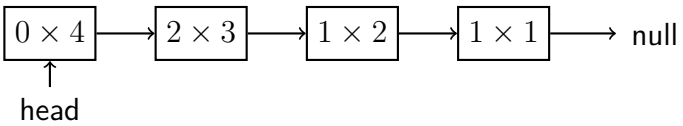
A multi-set is a set that can hold elements multiple times. The order of the elements in a set is irrelevant.
In the following implementation a multiset is stored as linked list where each element additionally contains the number of occurences in the set. Depicted in the following is the data structure corresponding to a single element of the set.

```
class ElementNode{
    ElementNode next;
    int value;
    int quantity;

    ElementNode (int v, ElementNode n){
        value = v;
        next = n;
        quantity = 1;
    }
}
```

Nachfolgend dargestellt ist der Inhalt der Multimenge $\{3, 3, 2, 1\}$, welche folgendermassen entstanden ist: Einer initial leeren Liste wurden die Elemente $1, 2, 3, 2, 3, 4$ hinzugefügt und nachfolgend wurden die Elemente 2 und 4 je einmal entfernt.
Zu erkennen ist auch, dass beim Entfernen eines Elementes aus der Multimenge dieses nicht aus der Liste genommen werden muss, sondern dass das Herunterzählen des Zählers in unserer Implementation genügt. Natürlich darf der Zähler nicht negativ werden.

Presented in the following is the content of the multi-set $\{3, 3, 2, 1\}$, which has been generated in the following way: to an initial empty list have the elements $1, 2, 3, 2, 3, 4$ been added, and subsequently the elements 2 and 4 have been removed once each.
It becomes apparent that when an element is removed from the multi-set, it does not have to be removed from the list but the counter only has to be decreased. The counter must not become negative, of course.



Ergänzen Sie den Code auf der rechten Seite, so dass die Datenstruktur `MultiSet` erwartungs- und dokumentationsgemäss funktioniert.

Complement the code on the right hand side such that the data structure `MultiSet` *would work as expected and documented.*

```
class MultiSet{
    ElementNode head=null;

    // Insert value into the multi-set. If the corresponding element exists
    // then increase its quantity, otherwise add a new element somewhere
    // to the list.
    public void insert(int value){
        ElementNode e = head;
        while (e != null && e.value != value){
            e = e.next;
        }
        if (e == null){
            
        } else {
            ++e.quantity;
        }
    }

    // Remove value from the multi-set.
    // If the element is contained in the set, then decrease its quantity.
    // Elements are at not physically removed from the list.
    public void remove(int value){
        ElementNode e = head;
        while (e != null && e.value != value){
            e = e.next;
        }
        if (                                                    ){
            e.quantity--;
        }
    }

    // Output all elements providing a quantity > 0.
    public void out(){
        for(                                                                ){
            if (n.quantity > 0){
                System.out.println(n.quantity + " x " + n.value);
            }
        }
    }
}
```

Aufgabe 4: Klassen, Kapselung und Floating Points (10P)

/6P (a) *Aufgabe aus dem Buch (gekürzt)*

Komplexe Zahlen. Eine komplexe Zahl (z.B. $3.2+1.75\mathbf{i}$) besteht aus einem reellen und einem imaginären Teil, beide vom Typ `float`. Implementieren Sie eine Klasse `Complex`, die komplexe Zahlen darstellt. Als Operationen sollen die Grundrechenarten sowie ein geeigneter Konstruktor angeboten werden.

Complex Numbers. A complex number (e.g. $3.2 + 1.75\mathbf{i}$) consists of a real and imaginary part, both of type `float`. Implement a class `Complex` that represents complex numbers. Operations should be the basic arithmetics as well as a suitable constructor.

Ergänzen Sie das vorgegebene Gerüst für die Klasse `Complex` in den dafür vorgesehenen grauen Boxen. Denken Sie daran, die Daten zu kapseln.

Complete the given skeleton for the class `Complex` using the designated gray boxes. Don't forget to encapsulate the data.

Tip aus dem Buch: $(a + b\mathbf{i}) + (c + d\mathbf{i}) = (a + c) + (b + d)\mathbf{i}$

```
public class Complex {
    // Fields
    private static final float epsilon = 0.00001; // floating point precision
    [                                                                          ]
    // Constructor
    [                                                                          ]
    [                                                                          ]
    /**
     * Adds the complex number 'other' to 'this'.
     * Modifies the current object to contain the result of the operation.
     */
    public void add(Complex other){
        [                                                                      ]
    }
    // Other operations omitted
    [...]
```

(b) Wir wollen in der Klasse `Complex` eine zusätzliche Methode `equals` anbieten, welche bestimmt, ob eine übergebene komplexe Zahl (`other`) "gleich" der aktuellen Zahl (`this`) ist. Implementieren Sie die untenstehende Methode `equals`.
Sie können zur Vereinfachung im Rahmen dieser Aufgabe davon ausgehen, dass die Beträge von Real- und Imaginärteil der beiden zu vergleichenden Zahlen in der Nähe von 1, also weder sehr klein noch sehr gross sind.

Tip: $|x|$ in Java: `Math.abs(x)`

We want to add an additional method `equals` *in class* `Complex` *that determines whether a provided complex number (*`other`*) is "equal" to the current number (*`this`*). Implement the method* `equals` *below.*
In the contect of this task you can assume that the absolute values of real and imaginary parts of the two numbers to be compared are close to 1*, i.e. neither very small nor very large.*

/4P

```
[...] // see left page for fields of class Complex

/**
 * Checks if 'other' and 'this' can be considered equal.
 * @return true if considered equal, false otherwise.
 */
public boolean equals(Complex other){
    
}
}
```

Aufgabe 5: Vererbung und Polymorphie (10P)

```
public abstract class Animal {
    private String movement;
    private String name;

    protected Animal(String name, String movement){
        this.name = name; this.movement = movement;
    }

    public void moveTo(String destination){
        Out.println(name + " " + movement + " to " + destination);
    }

    public abstract void fart();
}
```

```
public class Horse extends Animal {
    protected Horse(String name, String movement){
        super(name, movement);
    }

    public Horse(String name){
        super(name, "gallops");
    }

    public void fart(){
        Out.println("It smells bad!");
    }
}
```

```
public class Unicorn extends Horse {
    public Unicorn(String name){
        super(name,"flies");
    }

    @Override
    public void moveTo(String destination){
        Out.println("Scatter sparks!");
        super.moveTo(destination);
    }

    public void fart(){
        Out.println("A rainbow!");
    }
}
```

(a) Geben Sie für die folgenden Codeabschnitte an ob sie kompilieren. Wenn ja, schreiben Sie hin, was der Code ausgibt. Wenn nein, erklären Sie, warum nicht.

Indicate for the following code fragments if they compile. If yes, write down the output of the code. If no, explain why not.

/10P

```
Horse h = new Horse("Fury");
h.moveTo("town");
```

◯ Ausgabe / *Output*: ◯ Compiler Error:

```
Horse h = new Horse();
h.fart();
```

◯ Ausgabe / *Output*: ◯ Compiler Error:

```
Unicorn u = new Unicorn("Mystique");
Horse h = u;
h.fart();
```

◯ Ausgabe / *Output*: ◯ Compiler Error:

```
Horse h = new Unicorn("Mystique");
Unicorn u = h;
u.fart();
```

◯ Ausgabe / *Output*: ◯ Compiler Error:

```
Animal a = new Unicorn("Mystique");
a.moveTo("forest");
```

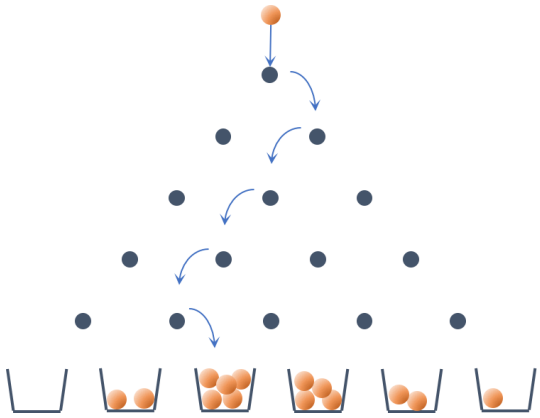
◯ Ausgabe / *Output*: ◯ Compiler Error:

Aufgabe 6: Matlab (4P)

Dies sollte Ihnen bekannt vorkommen: Eine Implementierung des Galton Boards. Statt mit `rand` zufällige Fliesskommazahlen zu erzeugen, verwenden wir die Funktion `randi([from, to])`, um eine zufällige Ganzzahl im Intervall [from, to] zu generieren (gleichverteilt).

This should look familiar: An implementation of the galton board. Instead of creating random floating point numbers with `rand`, we use the function `randi([from, to])` to create a random integer number in the interval [from, to] (uniformly distributed).

```
1   height = 5;
2   buckets = zeros(height+1,1);
3   for k=1:10000
4       sum=0;
5       for i=1:height
6           random = randi([1,10]);
7           if random > 5
8               sum=sum+1;
9           end
10      end
11      buckets(sum) = buckets(sum)+1;
12  end
```



/1P (a) Leider gibt es folgenden Fehler bei der Ausführung. Korrigieren Sie den obigen Code entsprechend (direkt im Code an einer einzigen Stelle).

Unfortunately, the following error occurs upon execution. Fix the code above appropriately (directly in the code at one place only).

```
Subscript indices must either be real positive integers or logicals.
Error in discrete_galton (line 11)
```

/1P (b) Da der Code nun funktioniert: Wie gross ist die Matrix `buckets` am Ende der Programmausführung?

As the code is working now: What is the size of the matrix `buckets` at the end of the program execution?

Dimension: [ ] × [ ]

/2P (c) Da Sie nun die Funktion `randi` kennen, können Sie die Zeilen 6-9 durch eine einzige Zeile ersetzen, ohne dass sich das Programmverhalten ändert:

As you are now aware of the function `randi`, replace lines 6-9 by a single line of code without changing the behaviour of the program:

Alternative Implementation (6-9): [ ]