Informatik II D-BAUG, Master 11.8.2014 *F. Friedrich*

Name, Vorname: .....

Legi-Nummer: .....

Ich bestätige mit meiner Unterschrift, dass ich diese Prüfung unter regulären Bedingungen ausführen 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:

a) Dauer der Prüfung: 120 Minuten.

b) Erlaubte Hilfsmittel: Zusammenfassung der Vorlesung auf maximal 8 A4-Seiten.

c) Benützen Sie einen Kugelschreiber (blau oder schwarz) und keinen Bleistift. Bitte schreiben Sie leserlich. Nur lesbare Resultate werden bewertet.

d) Lösungen sind direkt auf das Aufgabenblatt in die dafür vorgesehenen Boxen zu schreiben. Ungültige Lösungen deutlich durchstreichen. Korrekturen bei Multiple-Choice Aufgaben unmissverständlich anbringen.

e) Falls Sie sich durch irgendjemanden oder irgendetwas gestört fühlen, melden Sie dies sofort der Aufsichtsperson.

f) 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.

g) Wenn Sie zur Toilette müssen, melden Sie dies einer Aufsichtsperson durch Handzeichen. Es darf zu einer Zeit immer nur eine Studentin oder ein Student zur Toilette.

h) Wir beantworten keine inhaltlichen Fragen während der Prüfung. Kommentare zur Aufgabe schreiben Sie bitte auf das Aufgabenblatt.

General guidelines:

Exam duration: 120 minutes.

Permitted examination aids: lecture summary on maximally 8 A4 pages.

Use a pen (black or blue), not a pencil. Please write legibly. We will only correct solutions that we can read.

All solutions must be written directly onto the exercise sheets in the provided boxes. Invalid solutions need to be crossed out clearly. Corrections to answers of multiple choice questions must be provided without any ambiguity.

If you feel disturbed by anyone or anything, immediately let the supervisor of the exam know this.

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 preliminarily is only possible until 15 minutes before the exam ends.

If you need to go to the toilet, raise your hand and wait for a supervisor. Only one student can go to the toilet at a time.

We will not answer any content-related questions during the exam. Please write comments referring to the tasks on the exam sheets.

Aufgabe	1	2	3	4	5	6	7	8	9	10	$\sum$
Punkte											
Maximum	10	10	10	10	10	10	10	10	10	10	100

1 Klassendiagramme, Java Website *(10 Punkte)*

⊛ 6

a) Kennzeichnen Sie die folgenden Behauptungen als *wahr* (**w**) oder falsch (**f**). Wenn Sie zu einer Behauptung gar keine Antwort wissen, lassen Sie die entsprechende Box besser leer. Blindes Raten lohnt sich nicht, da falsche Antworten innerhalb dieser Aufgabe Abzug geben.

*Mark the following statements true (**w**) or false (**f**). If you do not know the answer for a statement, leave the box blank. Guessing blindly will not pay off since within this question you will get negative points for wrong answers.*

W	Das Klassendiagramm hilft umfangreichere Java-Projekte übersichtlich zu gestalten.	*A class diagram helps you to structure comprehensive java projects clearly.*
f	Grundsätzlich muss **jede** Java-Klasse in einer eigenen Datei stehen.	*In general **every** Java class needs to be saved in its own file.*
W	Wenn C *is-a* B und B *is-a* A gilt, so gilt immer auch C *is-a* A (Vererbung "*is-a*" ist transitiv).	*If C is-a B and B is-a A, then C is-a A always holds (Inheritance "is-a" is transitive).*
W	Vererbung erlaubt die Wiederverwendung von Code in abgeleiteten Klassen.	*Inheritance allows the reuse of code in subclasses.*
f	Polymorphie impliziert: eine einzige Instanz einer Klasse kann, je nach Aufrufer einer Methode, unterschiedlichen Code ausführen.	*Polymorhpism implies: a single instance of a class can, depending on the caller of a method, execute different code.*
W	Polymorphie impliziert: verschiedene Instanzen vom gleichen statischen Typ können beim Aufruf einer Methode unterschiedlichen Code ausführen.	*Polymorphism implies: different instances of the same static type can execute different code when a method is called.*

b) Zeichnen Sie das Klassendiagramm für folgenden Anwendungsfall. Sie müssen weder Variablen noch Methoden notieren.

Draw the class diagram for the following use case. You do not have to name variables or methods.

Jede Webseite werde von einer Java-Klasse gebildet. Es gibt jedoch auch Klassen, die selber keine Internetseite darstellen. Der Internetauftritt eines Vereins sei folgendermassen strukturiert:

Each web-site is formed by its own java class. But there are also classes which are not a webpage by itself. Let the internet presence of an association be structured as follows:

- Page definiert alles Grundlegende zur Darstellung der Internetseite, alle Webseiten leiten von dieser Klasse ab.

  Page defines the basics to show a web page. Every website inherits from this class.

- Die Hilfsklasse `Tool` wird von `Page` verwendet (*has-a*).

  The helper class `Tool` is used (has-a) by `Page`.

- Die Hilfsklasse `Directories` kann Verzeichnisstrukturen verarbeiten.

  The helper class `Directories` can be used to process directory structures.

- `Gallery` ist eine Internetseite, die die Klasse `Directories` verwendet.

  `Gallery` is a website which uses the class `Directories`.

- Die Hilfsklasse `Database` stellt die Anbindung zur Datenbank bereit.

  The helper class `Database` offers a connection to a database.

- `News` und `Content` leiten von `Page` ab und verwenden eine Datenbankanbindung.

  `News` and `Content` derive from `Page` and use a database connection.

⊛ 4

Ende Aufgabe 1

2 Ringspeicher *(10 Punkte)*

Betrachten Sie folgende Java-Klasse.

Consider the following Java class.

```
public class RingStorageMovingAverage {
    public final int SIZE = 32;
    private int[] ring;
    private int sum;
    private int pos;

    public RingStorageMovingAverage() {
        sum = 0;
        pos = 0;
        ring = new int[SIZE];
        for (int i=0; i<SIZE; ++i) {
            ring[i] = 0; // init
        }
    }

    public void add(int n) {
        sum -= ring[pos];
        sum += n;
        ring[pos] = n;
        pos = (pos + 1) % SIZE; // Modulo
    }

    public double average() {
        return (double)sum / SIZE;
    }
}
```

Kennzeichnen Sie die folgenden Behauptungen als *wahr* (**w**) oder falsch (**f**). Wenn Sie zu einer Behauptung gar keine Antwort wissen, lassen Sie die entsprechende Box besser leer. Blindes Raten lohnt sich nicht, da falsche Antworten innerhalb dieser Aufgabe Abzug geben.

*Mark the following statements true (**w**) or false (**f**). If you do not know the answer for a statement, leave the box blank. Guessing blindly will not pay off since within this question you will get negative points for wrong answers.*

⊛ 1p each

W	Sämtliche Methoden dieser Klasse sind von ausserhalb der Klasse aufrufbar.	*All methods of the given class can be called from the outside.*
f	Sämtliche Eigenschaften (Variablen) dieser Klasse sind von ausserhalb der Klasse lesbar.	*All properties (variables) of this class can be read from the outside.*
W	Es handelt sich um eine Art fortlaufende Durchschnittsberechnung.	*The class implements some sort of an average computation.*
f	Das Programm macht das Gleiche wie der in der Vorlesung gezeigte Algorithmus "Provisional Means". Das heisst jeder jemals hinzugefügte Wert beeinflusst den Durchschnitt bis zum Ende der Laufzeit.	*The program does the same as the algorithm "Provisional Means" from the lecture. This means that every value once added affects the average until the end of runtime.*
W	Um die Grösse "`SIZE`" anzupassen, muss das Programm neu kompiliert werden.	*To adapt the value of "`SIZE`", the program needs to be re-compiled.*
f	Da jeder neue Wert direkt Eingang in die gespeicherte Summe "`sum`" findet, bräuchten wir gar keinen Ringspeicher.	*Since every new value is added to the stored `sum` directly, we actually don't need the ring storage.*
f	Die Eigenschaft (Variable) "`pos`" kann nach zu vielen Einfüge-Operationen den zulässigen Bereich überschreiten und so zum Absturz führen.	*The property (variable) "`pos`" might exceed the legal range after too many add operations and thus crash the program.*
W	Eine Gleitkomma-Division ist relativ rechenintensiv. Wenn die Methode "`average`" öfter aufgerufen wird als "`add`", würde es sich deswegen lohnen die Division gerade am Ende von "`add`" durchzuführen und das Resultat für die Aufrufe von "`average`" zwischenzuspeichern.	*A floating point division is computationally relatively expensive. If the methode "`average`" is used more often than "`add`", it would pay off to move the division from "`average`" to the end of "`add`" and cache the result for calls of "`average`".*
W	Der Code "`return (double)(sum / SIZE);`" enthält keine Gleitkomma-Division.	*The code* *"`return (double)(sum / SIZE);`"* *does not contain a floating-point division.*
f	Der Code "`return (double)(sum / SIZE);`" liefert das selbe Resultat wie "`return (double)sum / SIZE;`".	*The code* *"`return (double)(sum / SIZE);`"* *yields the same result as* *"`return (double)sum / SIZE;`".*

Ende Aufgabe 2

3 Datenbanken *(10 Punkte)*

a) Zur Beispiellösung der VierGewinnt-Übung gab es vereinzelt Beschwerden. Wir haben uns deswegen überlegt einen Teil der Datenstruktur entsprechend allgemeiner zu gestalten. Bitte zeichnen Sie das vollständige ER-Diagramm mit Funktionalitäten passend zu den untenstehenden Datenbank-Tabellen.

There were some complaints regarding our sketched solution of the four-in-a row exercise. Therefore we have thought about a more general re-formulation of a part of the data structure. Please draw the complete ER-diagram with functionalities that fits the data base tables below.

Users

UserID	**Name**
1	Anna
2	Berta
3	Chiara
4	David

Play

RID	**UserID**	**GameID**
1	1	1
2	2	1
3	2	3
4	3	3

Games

GameID	**Timestamp**
1	102
2	243
3	378
4	378

⊛ 5

b) Betrachten Sie die folgenden Datenbank-Anfragen und schreiben Sie die entsprechenden Nummern zu den nachfolgenden Aussagen. Falsche Antworten geben in dieser Aufgabe keinen Abzug. Für jede Frage gilt der gleiche Startzustand wie in den obenstehenden Tabellen abgebildet ist.

Consider the following data base queries and write the corresponding numbers at the following statements. Wrong answers will not be punished with point deduction in this task. For each single question assume the same initial state as provided in the tables above.

⊛ 5:1/2 each

1. `SELECT COUNT(Name) FROM Users`
2. `SELECT DISTINCT Timestamp FROM Games WHERE Timestamp > 250`
3. `SELECT Name FROM Users ORDER BY Name DESC LIMIT 3`
4. `SELECT MIN(Timestamp) FROM Games`
5. `SELECT * FROM Users WHERE Name LIKE '%a'`
6. `SELECT GameID FROM Games WHERE Timestamp BETWEEN 200 AND 300`
7. `INSERT INTO Play (UserID, GameID) VALUES (4, 4)`
8. `UPDATE Users SET Name = 'Davide' WHERE UserID = 4`
9. `DELETE FROM Games WHERE Timestamp < 300`
10. $\pi_{Timestamp}(\sigma_{Name='Berta'}(Users) \bowtie Play \bowtie Games)$

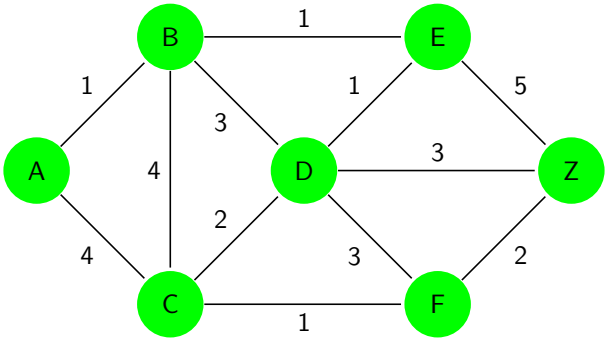
10	*Resultat:* 102, 378	*Result: 102, 378*
3	*Resultat:* David, Chiara, Berta	*Result: David, Chiara, Berta*
7	David wird zu Spiel 4 hinzugefügt.	*David is added to game 4.*
2	*Resultat:* 378	*Result: 378*
9	Die ersten beiden Zeilen in Games werden entfernt.	*The first two rows of Games are removed.*
1	*Resultat:* 4	*Result: 4*
8	Der Name von David wird geändert.	*David's name is modified.*
6	*Resultat:* 2	*Result: 2*
5	Es werden die ersten drei Zeilen der Tabelle Users ausgegeben.	*The first three rows of table Users are output.*
4	*Resultat:* 102	*Result: 102*

Ende Aufgabe 3

4 Kürzester Pfad *(10 Punkte)*

Im folgenden ungerichteten Graphen soll die kürzeste Strecke von A nach Z gefunden werden. Die Gewichte an den Kanten bezeichnen die Wegstrecke. Die Darstellung des Graphen spiegelt die angegebenen Weglängen nicht wieder.

In the following undirected graph the shortest path from A to Z shall be determined. The weights at the edges indicate the path lengths. The geometry of the graph does obviously not represent the provided lengths.



a) Wie lang ist der kürzeste Pfad von A nach Z? *How long is the shortest path from A to Z?* 6

⊛ 1

b) Geben Sie den kürzesten Pfad von A nach Z an: *Specify the shortest path from A to Z.*
ABEDZ

⊛ 2

c) Geben Sie die Sequenz der Punkte so an, wie sie zu der Menge der kürzesten Pfade im Algorithmus von Dijkstra hinzugenommen werden: *Specify the sequence of points as they are added to the set of shortest paths in Dijkstra's algorithm.*
ABEDCFZ

⊛ 3

d) Markieren Sie (nur), was richtig ist: *Mark (only) what is correct:*
Die Datenstruktur Min-Heap bietet sich im Algorithmus von Dijkstra an für die Repräsentation *The Min-Heap data structure is suitable in the context of Dijkstra's algorithm for*

⊛ 1

- [ ] der Menge G aller Punkte des gesamten Graphen. *the set G of all points of the whole graph.*
- [ ] der Menge Ω der Punkte, die bereits einem kürzesten Weg zugeordnet sind. *the set Ω of points that have already been assigned to a shortest path.*
- [ ] der Menge R der Punkte, die in G, aber nicht in Ω liegen. *the set R of points in G that are not in Ω.*
- [√] der Menge $\delta\Omega$ der Punkte in R, die auf dem Rand von Ω liegen. *the set $\delta\Omega$ of points in R on the boundary of Ω.*
- [ ] der Menge δR der Punkte in Ω, die auf dem Rand von R liegen. *the set δR of points in Ω on the boundary of R.*

⊛ 3

Wir haben Heaps im Zusammenhang mit verschiedenen Algorithmen kennengelernt. Sei im folgenden n die Anzahl der gespeicherten Elemente im Heap. Markieren Sie (nur) richtige Aussagen.

We have come across Heaps in the context of different algorithms. Let in the following n be the number of stored elements in the Heap. Mark (only) correct statements.

- ☐ Ein Min-Heap erlaubt die schnelle Berechnung des Minimums einer beliebigen mathematischen Funktion.
  A Min-Heap permits the fast computation of a minimum of an aribitrary mathematical function.
- √ Eine Kombination von Min- und Max-Heap kann in einem Online-Algorithmus zur Berechnung des Medians verwendet werden.
  A combination of Min- and Max-Heap can be used in an online-algorithm for the computation of the median.
- ☐ Eine Kombination von Min- und Max-Heap eignet sich für einen Online-Algorithmus zur Berechnung des Mittelwerts.
  A combination of Min- and Max-Heap is well suited for an online-algorithm for the computation of the mean.
- √ Die Extraktion des Minimums in einem Min-Heap ist eine schnelle Operation und benötigt maximal $O(\log n)$ Schritte.
  The extraction of the minimum in a Min-Heap is a fast operation and requires a maximum of $O(\log n)$ steps.
- √ Das Einfügen eines beliebigen Wertes in einen Min-Heap ist eine schnelle Operation und benötigt maximal $O(\log n)$ Schritte.
  To insert an arbitrary value in a Min-Heap is a fast operation and requires a maximum of $O(\log n)$ steps.
- ☐ Das Finden eines beliebigen Schlüssels in einem Min-Heap ist eine schnelle Operation und benötigt maximal $O(\log n)$ Schritte.
  To find an arbitrary key in a Min-Heap is a fast operation and requires a maximum of $O(\log n)$ steps.

Ende Aufgabe 4

5 Baum Traversieren *(10 Punkte)*

a) Im folgenden sehen Sie einen Teil der Implementation eines binären Suchbaumes. Gehen Sie davon aus, dass der nicht dargestellte Teil der Implementation eines binären Baumes korrekt ist. Stellen Sie die beiden Prozeduren `HasKeyIterative` und `HasKeyRecursive` so fertig, dass sie zurückgeben, ob der Unterbaum einen Knoten mit gegebenen Schlüssel enthält.

In the following you see a part of the implementation of a binary search tree. Assume that the non displayed part of the implementation of a binary tree is correct. Complement the two procedures `HasKeyIterative` *and* `HasKeyRecursive` *so that they return if the sub-tree contains a node with the provided key.*

```
class SearchNode {
    int key;
    SearchNode lower;
    SearchNode higher;
    // ...
}

// returns if the sub-tree starting at root contains a node with given key
public boolean HasKeyIterative(SearchNode root, int key) {
    SearchNode node = root;
    while (node != null && node.key != key)
    {
        if (key < node.key)    node = node.lower;
        else                   node = node.higher;
    }
    return node != null;
}

// returns if the sub-tree starting at root contains a node with given key
public boolean HasKeyRecursive(SearchNode root, int key) {
    if (root == null) return false;
    else if (key == root.key) return true;
    else if (key < root.key) return HasKeyRecursive(root.lower, key);
    else                     return HasKeyRecursive(root.higher, key);
}
```

⊛ 3

⊛ 3

b) Angenommen, der iterative und rekursive Algorithmus werden auf denselben Baum mit dem selben key angewendet. Natürlich sollten beide Algorithmen dasselbe Resultat liefern. Aber wie ist es mit der Qualität gemessen in Anzahl Schlüsselvergleichen? Markieren Sie (nur) richtiges:

Assume the iterative and recursive algorithm are applied to the identical tree with the same key. Of course both algorithms are assumed to deliver the same result. But what is about their quality measured in number of key comparisons? Mark (only) what is correct:

⊛ 1

- [√] Die Anzahl Schlüsselvergleiche des iterativen und rekursiven Algorithmus sind gleich.

  The number of key comparisons of the iterative and recursive algorithms equal

- [ ] Die Anzahl Schlüsselvergleiche des iterativen Algorithmus ist grösser als die des rekursiven Algorithmus.

  The number of key comparisons of the iterative algorithm is greater than that of the recursive algorithm.

☐ Die Anzahl Schlüsselvergleiche des iterativen Algorithmus ist kleiner als die des rekursiven Algorithmus.

The number of key comparisons of the iterative algorithm is smaller than that of the recursive algorithm.

☐ Das hängt vom Inhalt des Baumes ab.

It depends on the content of the tree.

☐ Das hängt vom Schlüsselwert ab.

It depends on the value of the key.

c) Geben Sie die zeitliche asymptotische worst-case Komplexität der Algorithmen in Abhängigkeit von der Anzahl Knoten n eines **nicht balancierten** binären Suchbaumes an:
$O(n)$ (für beide)

Provide the asymptotic computational worst-case complexity of the algorithms with respect to the number of nodes n of a **non-balanced** *binary search tree.*

⊛ 1

Geben Sie die zeitliche asymptotische worst-case Komplexität der Algorithmen in Abhängigkeit von der Anzahl Knoten n eines **balancierten** binären Suchbaumes an:
$O(\log n)$ (für beide)

Provide the asymptotic computational worst-case complexity of the algorithms with respect to the number of nodes n of a **balanced** *binary search tree.*

⊛ 1

d) Geben Sie den asymptotischen worst-case **Speicherplatzverbrauch der Algorithmen** `HasKeyIterative` und `HasKeyRecursive` in Abhängigkeit von der Anzahl Knoten n eines **balancierten** binären Suchbaumes an:
$O(1)$ (iterativ), $O(\log n)$

Provide the asymptotic worst-case **memory consumption of the algorithms** `HasKeyIterative` *and* `HasKeyRecursive` *with respect to the number of nodes n of a* **balanced** *binary search tree.*

⊛ 1

Ende Aufgabe 5

6 Programmausgaben *(10 Punkte)*

Geben Sie die Ausgaben von `main` jeweils im dafür vorgesehenen Kasten an.

Provide the output of `main` *for each example in the designated boxes below.*

a)
```
public static int R(int i) {
    if (i>1)
        return R(i-1) * i;
    else
        return 1;
}

public static int I(int i) {
    int res = 0;
    while (i>0) {
        res += (i) * (i-1);
        --i;
    }
    return res;
}

public static void main(String[] args) {
    System.out.println( R( 6 ) );
    System.out.println( I( 6 ) );
}
```

720
70

⊛ 4=2+2

b)
```
public static void main(String[] args) {
    String a[] = new String[2];
    String b[] = a;
    String c[] = a;
    a[0] = "Hund";
    a[1] = "Katze";
    b[0] = "Maus";
    c = new String[2];
    c[1] = "Elefant";
    for (int i = 0; i<a.length; ++i)
        System.out.println(a[i]);
}
```

Maus
Katze

⊛ 3=2+1

c)
```
class Person {
    String name;

    Person(String n){ name = n; }

    void Opening(){
        System.out.println("Hello " + name);
    }
}

class Man extends Person {
    Man(String n){ super(n); } // super (n) is a call to Person(n) in superclass

    void Opening(){
        System.out.println("Dear Mr. " + name);
    }
}

public class Test {
    public static void Letter(Person s) {
        s.Opening();
        // rest of letter text, omitted here for brevity
    }

    public static void main(String[] args) {
        Man brad = new Man ("Pitt");
        Person mickey = new Man ("Maus");
        Person lucky = new Person ("Luke");

        Letter(brad);
        Letter(mickey);
        Letter(lucky);
    }
}
```

Dear Mr. Pitt
Dear Mr. Maus
Hello Luke

⊛ 3=1+1+1

Ende Aufgabe 6

7 Fehlersuche *(10 Punkte)*

In jedem der nachfolgenden Programme steckt ein Fehler. Sämtliche Beispiele kompilieren vollständig. Die vorhandenen Fehler werden vom Compiler also **nicht** bemerkt. Vermerken Sie jeweils kurz den Fehler und dessen Auswirkung in den Boxen unter den Programmen. In dieser Aufgabe betrachten wir Code dann als fehlerfrei, wenn er korrekt terminiert.

*Each of the following programs has a bug. All examples compile successfully. That means the bug is **not** identified by the compiler. Please note down the bug and its consequence in the boxes below the programs. Within this task we consider code bug-free if it terminates correctly.*

a)
```
public int sum(int[] numbers) {
    int s = 0;
    for (int i=0; i<=numbers.length; ++i)
        s += numbers[i];
    return s;
}
public static void main(String[] args) {
    int a[] = {1,2,3};
    int s = sum(a);
    // ...
}
```

Fehler / *error*
Arraybereich wird überschritten, Arraygrenzen nicht eingehalten

Auswirkung / *effect*
Buffer Overrun, Exception, Crash

⊛ 2=1+1

b)
```
public int sqr(int n) {
    int r = 1;
    int a = 10000;
    while (a > 0) {
        while (r*r <= n)
            r += a;
        r -= a;
    }
    return r;
}
public static void main(String[] args) {
    int x = sqr (100);
    // ...
}
```

Fehler / *error*
Variable a wird nicht verändert & Endlosschleife

Auswirkung / *effect*
Programm terminiert nicht

⊛ 2=1+1

c)
```
public int log7(int x) {
    int d = x / 7;
    int n = log7(d);
    if (d >= 1) return 1+n;
    return 0;
}
public static void main(String[] args) {
    int x = log7 (100);
    // ...
}
```

Fehler / *error*
Rekursion ohne Abbruchbedingung

Auswirkung / *effect*
Stack Overflow (Crash)

⊛ 3=1+2

d)
```
public static int[] div(int[] a, int[] b) {
    for (int i=0; i<a.length; ++i)
        b[i] = a[i] / b[i];
    return b;
}

public static void main(String[] args) {
    int[] a = {0,1,2};
    int[] b = {3,4,5};
    int[] c = div(div(a,b),b); // c = a / b^2
    // ... print c ...
}
```

Fehler / *error*
Leaking, Parameter gets modified through reference, Division by zero

Auswirkung / *effect*
Division by zero, Crash, Exception

⊛ 3=1+2

Ende Aufgabe 7

8 Datenstrukturen *(10 Punkte)*

Gegeben seien folgende Daten, eine Sequenz von Buchstaben:

Consider the following data, a sequence of characters.

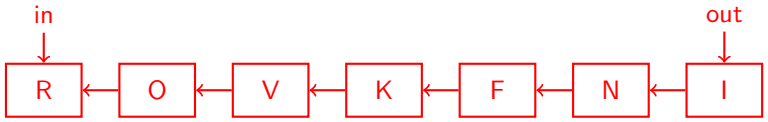
"I", "N", "F", "K", "V", "O", "R"

Vervollständigen Sie folgende Figuren, so dass sie deutlich machen, wie die jeweilige Datenstruktur nach Einfügen der gegebenen Daten in obiger Reihenfolge (von links nach rechts) aussieht. Vergessen Sie nicht, die Daten (Buchstaben) auch einzutragen.

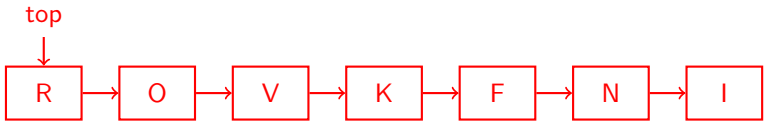
Complete the following figures such that they illustrate the respective data structures after insertion of the given data in the order above (from left to right). Do not forget to enter the data (characters) also.

a) Warteschlange (FIFO) mit verketteter Liste — *Queue (FIFO) with linked list*

⊛ 2

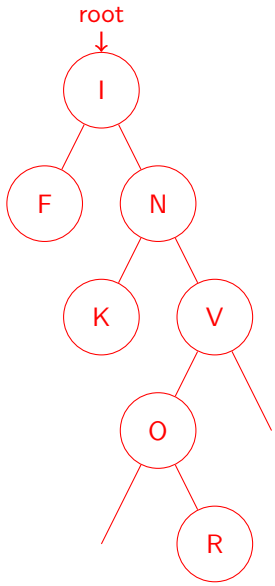


b) Stack mit verketteter Liste — *Stack with linked list*



⊛ 2

c) Binärer Suchbaum — *Binary search tree*

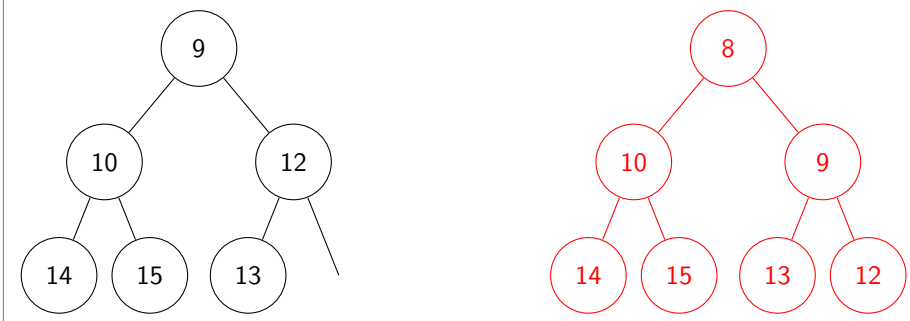


⊛ 2

(d) Zeichnen Sie neben folgendem Min-Heap den Min-Heap ein, welcher sich nach dem Algorithmus der Vorlesung ergibt, wenn man die Zahl 8 einfügt.

Draw next to the following Min-Heap the Min-Heap that results from the algorithm shown in the lectures after insertion of the number 8.

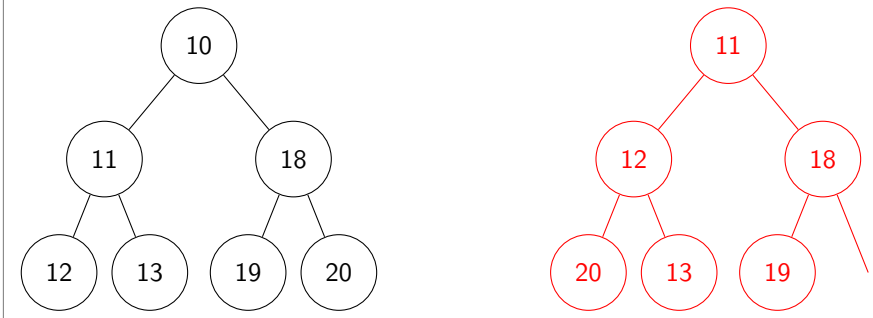
⊛ 2



(e) Zeichnen Sie neben folgendem Min-Heap den Min-Heap ein, welcher sich nach dem Algorithmus der Vorlesung ergibt, wenn man das Minimum entfernt.

Draw next to the following Min-Heap the Min-Heap that results from the algorithm shown in the lectures after removal of the minimum.

⊛ 2



Ende Aufgabe 8

9 Fussball *(10 Punkte)*

Das Bild zeigt den Spielplan der vergangenen Fussball-WM ohne Gruppenspiele. Es handelt sich um eine Baumstruktur. Dass sich der Baum hier in zwei Richtungen ausdehnt, ist rein graphisch bedingt.

The image shows the playing schedule of the recent football World Cup without any group games. It constitutes a tree structure. The fact that this tree streches into two directions, is caused graphical without any additional meaning.

Wir **formalisieren** die Spielabläufe, die Realität stark vereinfachend, wie folgt:

*We **formalize** the playing schedule, simplifying reality considerably, as follows:*

- Einer Mannschaft sei ein konstanter Punktwert zugeordnet.

  To each team there is a constant score assigned.

- Keine zwei Mannschaften haben den gleichen Punktwert.

  No two teams have the same score.

- Ein Spiel vergleicht die Punktwerte der beiden beteiligten Mannschaften. Die Mannschaft mit dem höheren Punktwert gewinnt das Spiel.

  A game compares the scores of the two teams involved. Always the team with the higher score wins.

Es stellen sich bzgl. der WM interessante Fragen zum Thema Selektion und Sortierung.

Some interesting questions arise concerning selection and sorting.

Beantworten Sie folgende Fragen, basierend auf den genannten Annahmen:

Answer the following questions, based on the given assumptions:

a) Wird der Sortiervorgang durch die WM vollständig abgeschlossen? Steht also schlussendlich fest, welche Mannschaft den z. B. siebten Platz belegt?

Does the world cup provide a complete sorting? Can therefore determined at the end which team is on the, say, seventh place?

☑ Sortierung ist **unvollständig**. *The sorting is **incomplete**.*

☐ Die Sortierung ist **vollständig**, den siebten Platz belegt: *The Sorting is **complete**, the seventh place goes to:*

⊛ 1

b) Argentinien belegt offiziell den zweiten Platz. Finden Sie ein (hypothetisches) Beispiel an Punktzahlen bei denen Frankreich besser ist als Argentinien und trotzdem sämtliche abgebildeten Spiele genau gleich ausgehen. Bitte tragen Sie die nötigen Zahlen direkt in die Boxen im Spielplan ein.

Argentina is officially placed second. Find a (hypothetical) example set of scores which does not change the result of any game but where the score of France is clearly higher than that of Argentina. Please write the numbers directly into the boxes in the playing schedule.

⊛ 1

c) Wie viele Spiele benötigt man minimal, wenn man in der Meisterschaft der 16 Mannschaften nur die beste Mannschaft ermitteln will?

How many games are minimally needed if during the championship of the 16 teams only the best team should be determined?

15

⊛ 1

d) Nun soll mit einem anderen Algorithmus eine Rangliste der 16 Mannschaften erstellt werden. Jemand schlägt vor, jede Mannschaft jeweils einmal gegen jede andere Mannschaft spielen zu lassen. Es ergibt sich eine Rangliste anhand der Anzahl erzielter Siege.

Now we wish to use another algorithm to determine the ranking of the 16 football teams. Somebody suggests that each and every team plays against each other team. The ranking is then given by the teams' numbers of winnings.

Dieses Sortierverfahren ist: *This sorting procedure is:*

☑ korrekt *correct*

☐ falsch *wrong*

⊛ 1

Wie viele Spiele (Vergleiche) finden statt? *How many games (comparisons) will take place?*

$\frac{(n-1)n}{2} = 120$

⊛ 2

e) Als weitere Alternative wollen wir **Bubblesort** als Sortiermechanismus betrachten.

*As another alternative we now would like to take a look at **Bubblesort** as a sorting mechanism.*

```
int NUM = 16;
int[] arr = new int[NUM];
// ... fill array with capabilites
int count = 0;
boolean done = false;
while (count < NUM-1 && !done) {
    done = true; // assume done
    ++count;
    for (int i=0; i<NUM-1; ++i) {
        if (arr[i] > arr[i+1]) {
            int t = arr[i];
            arr[i] = arr[i+1];
            arr[i+1] = t;
            done = false;
        }
    }
}
```

Wie viele Elementvergleiche finden **minimal** statt?

*How many comparisons are taking place **minimally**?*

⊛ 1

15

Wie viele Elementvergleiche finden **maximal** statt?

*How many comparisons are taking place **maximally**?*

⊛ 1

$15^2 = 225$

f) **Mergesort** ist oftmals schneller als Bubblesort, weil einmal gemachte Vergleiche nicht wiederholt werden. Denken wir uns für diese Aufgabe zwei bereits sortierte Listen, die nun zusammengefügt werden sollen. Die Ländernamen sind nicht aufgelistet, aber Sie können sich vorstellen, dass jede Zahl für eine Mannschaft steht.

***Mergesort** is often faster than Bubblesort, since comparisons are not repeated. Consider two sorted lists, which now need to be merged. The corresponding country names have been ommited but you may assume that every number stands for a football team.*

1
3
4
...

2
5
6
...

Wieviele Vergleiche sind **maximal** nötig, um zwei allgemeine, **sortierte** Listen der jeweiligen Länge 4 in eine neue, **sortierte** Liste mit der Länge 8 zu vereinigen?

*How many comparisons are **maximally** needed to merge two general, **sorted** lists, each of length 4, into a new, **sorted** list of length 8?*

⊛ 2

$(n-1) = 7$

Ende Aufgabe 9

10 Random Walk *(10 Punkte)*

Wir simulieren eine Irrfahrt eines Teilchens auf einem zweidimensionalen Gitter. In jedem Schritt soll das Teilchen entweder stehen bleiben oder in eine der vier Himmelsrichtungen vorrücken. Die Wahrscheinlichkeit für das Stehenbleiben sei 0.6 und für das Vorrücken in eine der vier Himmelsrichtungen jeweils 0.1. Bitte ergänzen Sie die Prozedur `Simulate` in untenstehendem Code so, dass eine entsprechende Irrfahrt ausgegeben wird.

Hinweis: `Math.random()` gibt eine uniformverteilte Zufallszahl im Intervall $[0, 1)$ zurück.

We simulate a random walk of a particle on a two dimensional grid. In each step the particle stays where it is or it moves in one of the four cardinal directions. Let the probability for standing still be 0.6 and for moving in one of the four cardinal directions 0.1 (each). Please complement the procedure `Simulate` in the code below so that the corresponding random walk is simulated.

Hint: `Math.random()` *yields a uniformly distributed random number in the interval* $[0, 1)$.

```
class Point{
    int x,y;
    Point (int x0, int y0)  {
        x = x0; y = y0;
    }
}

public class Simulation  {
    static final int NumSteps = 100; // constant number of steps

    static void Simulate(Point p){
        double r = Math.random();
        if (r<0.1) ++p.x
        else if (r<0.2) --p.x;
        else if (r<0.3) ++p.y;
        else if (r<0.4) --p.y;
        //else nothing ;
    }

    public static void main(String[] args) {
        Point point = new Point(0,0);
        for (int i = 0; i<NumSteps; ++i)
        {
            Simulate(point);
            System.out.println(point.x + " " + point.y);
            // code for visualisation omitted here for brevity
        }
    }
}
```

⊛ 10

Ende Aufgabe 10