

Informatik II D-BAUG Prüfung Lösung 26.1.2015 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:

- Dauer der Prüfung: 120 Minuten.
- Erlaubte Hilfsmittel: Zusammenfassung der Vorlesung auf maximal 8 A4-Seiten.
- Benützen Sie einen Kugelschreiber (blau oder schwarz) und keinen Bleistift. Bitte schreiben Sie leserlich. Nur lesbare Resultate werden bewertet.
- Lösungen sind direkt auf das Aufgabenblatt in die dafür vorgesehenen Boxen zu schreiben. Ungültige Lösungen deutlich durchstreichen. Korrekturen bei Multiple-Choice Aufgaben unmissverständlich anbringen.
- Falls Sie sich durch irgendjemanden oder irgendetwas gestört fühlen, melden Sie dies sofort der Aufsichtsperson.
- 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.
- 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.
- Wir beantworten keine inhaltlichen Fragen während der Prüfung. Kommentare zur Aufgabe schreiben Sie bitte auf das Aufgabenblatt.
- Es gilt generell in dieser Prüfung: Falsche Antworten ergeben keinen Punktabzug.

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.*
- Generally for this exam it holds that wrong answers do not lead to deduction of points.*

Aufgabe	1	2	3	4	5	6	7	8	Σ
Punkte									
Maximum	10	14	16	15	16	9	12	12	104

1 Klassendiagramm Robotersteuerung (10 Punkte)

Wir betrachten in dieser Aufgabe das Beispiel einer Robotersteuerung.

In this problem we consider a robot control unit as an example.



Folgende Designentscheidungen stehen bezüglich der objektorientierten Implementierung der Steuerung fest.

The following design decisions have been made with respect to the object-oriented implementation of the robot control.

- Die Software umfasst folgende Klassen: Sensors, Motors, Controller, Task, Circle, Square und Demo
- Roboter-Sensoren werden über die Klasse Sensors ausgelesen.
- Auf Roboter-Motoren wird über die Klasse Motors zugegriffen.
- Die Robotersteuerung wird in der Klasse Controller implementiert.
- Aufgaben für einen Roboter werden durch die Superklasse Task beschrieben.
- Die Klassen Circle, Square und Demo sind Implementationen der Superklasse Task.
- Controller hat Objekte vom Typ Sensors, Motors und Task.

The software comprises the following classes: Sensors, Motors, Controller, Task, Circle, Square and Demo

Robots sensors are accessible via class Sensors.

The robots motors are accessible via class Motors.

The robot control is implemented with the class Controller.

Tasks for the robot are described through the superclass Task.

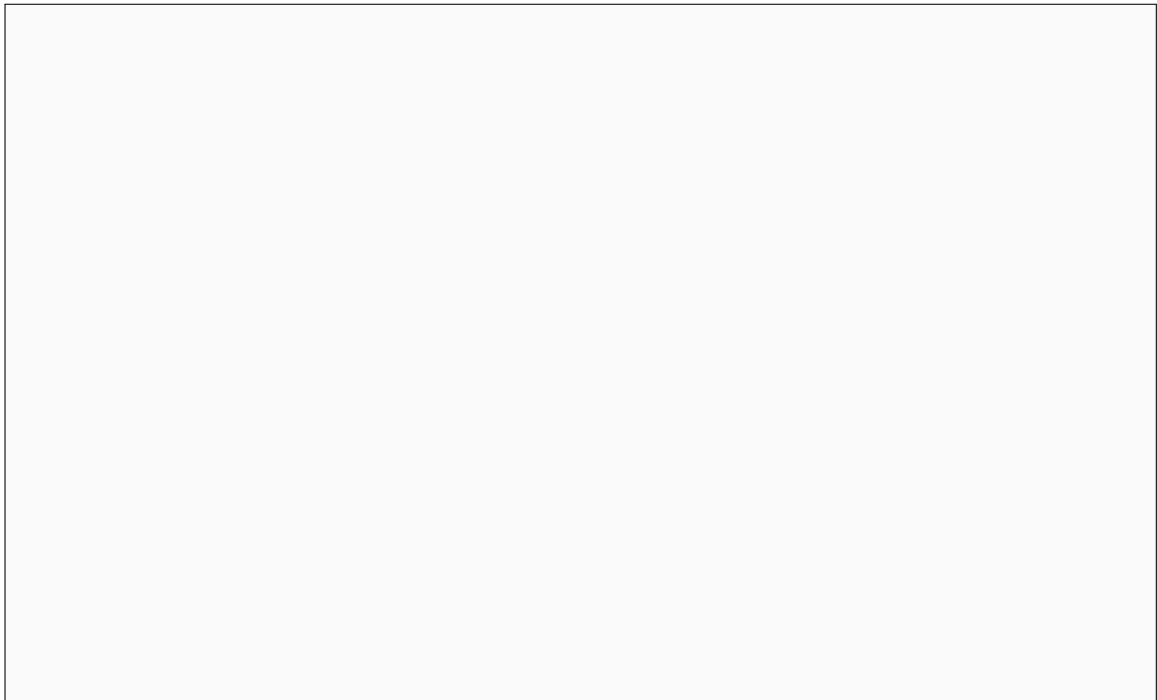
The classes Circle, Square and Demo are actual implementations of the superclass Task.

Controller has objects of type Sensors, Motors and Task.

-
- (a) Zeichnen Sie das Klassendiagramm für die beschriebene Robotersteuerung. Sie müssen weder Variablen noch Methoden notieren.

Draw the class diagram for the described robot control. You do not have to name variables or methods.

10 P



2 Datenspeicher (14 Punkte)

Betrachten Sie folgende Java-Klasse.

Consider the following Java class.

```
public class Storage
{   private int [] data;

    public Storage()
    {   data = new int [10]; // default size
    }

    private void resize(int minLength)
    {   int [] tmp = data;
        data = new int [minLength*3];
        for (int i=0; i<tmp.length; ++i)
        {   data[i] = tmp[i];
        }
    }

    public int get(int ind)
    {   if (ind >= data.length)
        {   return 0; // default number
        }
        return data[ind];
    }

    public void set(int ind, int val)
    {   if (ind >= data.length)
        {   resize(ind);
        }
        data[ind] = val;
    }
}
```

Kennzeichnen Sie die folgenden Behauptungen als wahr (w) oder falsch (f).

Mark the following statements true (w) or false (f).

(a) Wahr oder falsch? true or false?

W

2 P

Es handelt sich um eine Klasse die gegebene Daten intern in einem Array speichert.

The class stores given data internally in an array.

(b) Wahr oder falsch? true or false?

F

2 P

Um mittels der Methode set 20 Elemente im Array zu speichern muss die 10 auf Zeile 5 in mindestens eine 20 umgeändert werden und das Programm muss neu kompiliert werden.

For storing 20 elements in the array data using method set the number 10 in line 5 must be changed into 20 or more and the program has to be freshly compiled.

(c) Wahr oder falsch? true or false?

W

2 P

Wenn die Methode get über den Array-Bereich hinaus lesen soll, gibt sie einfach den Wert 0 zurück.

If method get shall read beyond the end of the array range it simply returns a value of 0.

(d) Wahr oder falsch? true or false?

F

2 P

Wenn die Methode set über den Array-Bereich hinaus schreiben soll, bricht das Programm mit einem Fehler ab.

When the method set shall write beyond the valid range of the array the program terminates with an error.

(e) Wahr oder falsch? true or false?

W

2 P

Wenn die Methode set über den Array-Bereich hinaus schreiben soll, ersetzt es einfach den Array durch einen grösseren Array.

When the method set shall write beyond the valid range of the array the array is simply replaced by a larger array.

(f) Betrachten Sie folgenden Code.

Consider the following code.

4 P

```
Storage s = new Storage ();  
s.set (3,1);  
s.set (11,2);  
s.set (20,34);  
s.set (33,1);
```

Geben Sie die resultierende Grösse des Arrays data im Storage s an.

Provide the resulting size of array data in Storage s.

99

3 Datenbanken (16 Punkte)

Gegeben sind folgende Tabellen.

Consider the following tables.

Stationen		
StationID	Ort	Höhe
1	Bellwald	1560
2	Fusio	1234
3	Bergün	1367
4	Eiger	3970
5	Eiger	3970

Messwerte				
ID	StationID	Timestamp	Temperatur	Wind
1	1	90	10	12
2	1	111	8	20
3	2	50	12	7
4	2	76	14	12
5	1	231	11	14
6	3	543	-2	30
7	4	250	-20	30
8	5	600	-32	0

8 P

(a) Formulieren Sie folgende Abfragen mit SQL.

Formulate the following queries using SQL.

Auflisten der Windgeschwindigkeiten aller Messwerte die eine Temperatur von unter 0°C angeben.

List windspeeds of measurement sets with a temperature below 0°C.

```
SELECT Wind FROM Messwerte WHERE Temperatur < 0
```

Auflisten von StationID aller Stationen, die unter 1500m ü.M. liegen.

List StationID of all stations that are located below 1500 m. above sea level

```
SELECT StationID FROM Stationen WHERE Stationen.Höhe < 1500
```

8 P

(b) Betrachten Sie die folgenden acht Datenbank-Anfragen.

Consider the following eight data base queries.

(A) SELECT DISTINCT Wind FROM Messwerte WHERE Timestamp < 100

(B) SELECT Ort FROM Stationen ORDER BY StationID DESC LIMIT 1

(C) SELECT StationID FROM Stationen WHERE Ort LIKE "B%"

(D) SELECT Temperatur FROM Messwerte WHERE Timestamp BETWEEN 100 AND 300

(E) INSERT INTO Stationen (StationID, Ort, Höhe) VALUES (6, "Glattalp", 1850)

(F) UPDATE Messwerte SET Temperatur = 5 WHERE StationID = 3

(G) DELETE FROM Messwerte WHERE Timestamp < 550

(H) $\pi_{Wind}(Messwerte \bowtie \sigma_{Hoehe > 1300m}(Stationen))$

Schreiben Sie die entsprechenden Buchstaben zu den nachfolgenden Aussagen. Es gibt Aussagen ohne dazugehörige Anfragen. Für jede Anfrage gilt der gleiche Startzustand wie in obigen Tabellen abgebildet.

E Fügt Glattalp den Messstationen hinzu

Korrigiert die StationID von Glattalp

B Resultat: Eiger

Resultat: Fusio

D Resultat: 8; 11; -20

Resultat: 8; 11; -2

G Löscht alle Messwerte mit Ausnahme des letzten

A Resultat: 12; 7

Resultat: 12; 7; 12

Resultat: Bellwald

C Resultat: 1; 3

H Resultat: 12,20,14,30,0

Resultat: 12,20,7,14,30,0

Korrigiert die Höhe von Glattalp

Löscht alle Messwerte

F Verändert Temperatur von Messwert mit ID=6

Write the corresponding characters at the following statements. There are statements without a mapping query. For each single question assume the same initial state as provided in the tables above.

Adds Glattalp to the stations

Changes the StationID of Glattalp

Result: Eiger

Result: Fusio

Result: 8; 11; -20

Result: 8; 11; -2

Deletes all values except the last one

Result: 12; 7

Result: 12; 7; 12

Result: Bellwald

Result: 1; 3

Result: 12,20,14,30,0

Result: 12,20,7,14,30,0

Changes the altitude of Glattalp

Deletes all values

Updates the temperature of measurement with ID=6

4 Verkettete Liste (15 Punkte)

Eine verkettete Liste sei wie folgt aus Elementen der Klasse `ListElement` aufgebaut. Das erste Element der Liste stehe in der Variable `LinkedList.first` zur Verfügung.

Consider the linked list below consisting of elements of class `ListElement`. The first element of the list is provided as variable `LinkedList.first`.

```
class ListElement{
    public int x;
    public ListElement next;
}

class LinkedList{
    private ListElement first = null; // reference of the first element

    public void print(){
        ListElement e = first;
        while ( e != null ) {
            System.out.println(e.x);
            e = e.next;
        }
    }

    public void append(int x){
        ListElement n = new ListElement();
        n.x = x;
        n.next = null;

        if ( first == null ){
            first = n;
        }
        else {
            ListElement e = first;
            while ( e.next != null ) {
                e = e.next;
            }
            e.next = n;
        }
    }
}
```


-
- (a) Vervollständigen Sie die Methode `print()`, so dass sie die Zahlen aus der ganzen Liste auf der Kommandozeile ausgibt. *Complete the method `print()` such that it outputs the numbers of the list on the command line via.* 6 P
-
- (b) Vervollständigen Sie die Methode `append(int x)`, so dass sie den Wert von `x` an das Ende der verketteten Liste anhängt. *Completes the method `append(int x)`, such that the value of `x` is appended at the end of the linked list.* 9 P

5 Suchbäume (16 Punkte)

Gegeben seien folgende String-Elemente:

Consider the following string elements.

Anna Berta Chiara Daniela Emma Foxy Gutemine

4 P

- (a) Zeichnen Sie einen vollständig balancierten binären Suchbaum, der diese 7 Elemente enthält.

Draw a fully balanced binary search tree that contains the 7 elements.

solution

2 P

- (b) Nennen Sie eine Reihenfolge, in der die genannten Elemente in einen binären Suchbaum eingefügt werden müssten, so dass dieser stets möglichst balanciert bleibt. Gehen Sie von einer naiven Implementierung ohne Balancierung aus. Sie können die Namen mit dem Anfangsbuchstaben abkürzen.

Provide an order in which the elements from above have to be filled in a binary search tree such that the tree constantly remains optimally balanced. Assume a naive implementation without balancing. You may abbreviate names by their first letter.

D C E B A E F

4 P

- (c) Ein binärer Baum enthalte 255 Elemente. Wie tief ist der Baum maximal?

Assume a binary tree containing 255 Elements. What is the maximal depth of the tree?

255

Ein binärer Baum enthalte 255 Elemente. Wie tief ist der Baum minimal?

Assume a binary tree containing 255 elements. What is the minimal depth of the tree?

8

(d) Wie viele Vergleichsoperationen sind maximal nötig, um ein Element in einer unsortierten verketteten Liste der Länge n zu finden?

How many comparisons are maximally required to find an element in an unsorted linked list with length n ?

6 P

$O(n)$

Wie viele Vergleichsoperationen sind maximal nötig, um ein Element in einer sortierten verketteten Liste der Länge n zu finden?

How many comparisons are maximally required to find an element in a sorted linked list with length n ?

$O(n)$

Wie viele Vergleichsoperationen sind maximal nötig, um ein Element in einem sortierten Array der Länge n zu finden?

How many comparisons are maximally required to find an element in a sorted array with length n ?

$O(\log_2 n)$

6 Vererbung (9 Punkte)

9 P

Betrachten Sie folgende Klassen A, B und C. Geben Sie in den vorgesehenen Kästen in der main-Methode des Main-Programmes die erzeugten Ausgaben der jeweiligen Zeile an.

Consider the following classes A, B and C. Provide in the provided boxes in the main method of the Main program the output generated by the respective line.

```
class A{

    int X(){
        return 1;
    }

    int Y(){
        return X();
    }
}

class B extends A {
    int Y(){
        return X()+10;
    }
}

class C extends B {
    int X(){
        return 100;
    }
}

public class Main {

    static void print(A a) {
        System.out.println(a.X() + "," + a.Y());
    }

    public static void main(String [] args) {

        print(new A()); // Ausgabe: 1 1

        print(new B()); // Ausgabe: 1 11

        print(new C()); // Ausgabe: 100 110

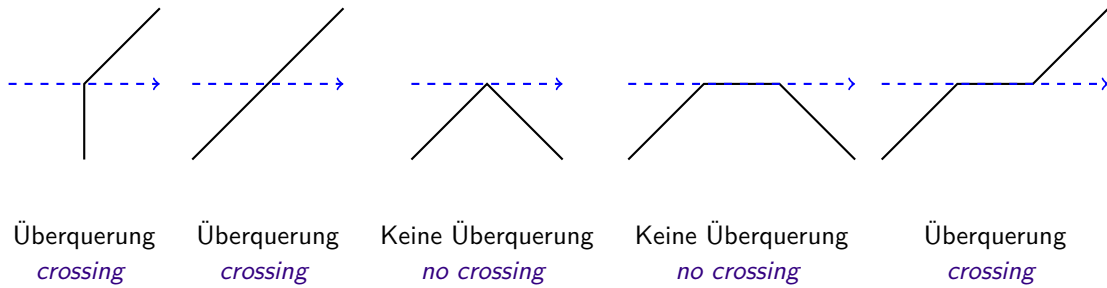
    }
}
```

Ende Aufgabe 6

7 Point In Polygon (12 Punkte)

Eine Ameise sitzt in einem grossen, rechteckigen Raum. Im gleichen Raum ist auf dem Boden ein Polygon eingezeichnet, welches keine Wand berührt. Die Ameise erkennt, wenn sie die Begrenzung des Polygons überschreitet. Die Ameise kann auch echte von unechten Überquerungen unterscheiden, vgl. Abbildungen unten.

An ant sits in a small rectangular room. In the same room on the floor a polygon is painted that does not hit any of the walls. The ant recognizes when it crosses the boundaries of the polygon. Moreover, the ant can distinguish between real from tangential crossings, cf. figures below.



4 P

- (a) Die Ameise läuft von einem Punkt p im Raum parallel zu einer der Wände los. Sie behält Ihre Laufrichtung ständig bei bis sie an eine Wand stösst. Dabei zählt sie n echte Überquerungen. Vervollständigen Sie folgende Methode, so dass sie `true` zurückgibt, genau dann wenn p im Inneren des Polygons liegt.

The ant starts running from a point p in the room parallel to one of the walls. It constantly keeps its direction until it hits a wall. Overall it counts n real crossings. Complete the following method such that it returns `true` if and only if p is in the inner of the polygon.

```
bool inPolygon (Point p)
{
    int n = IntersectionCount(p); // n = number of crossings of the ant
    return n % 2 == 1;
}
```

2 P

- (b) Wahr oder falsch? *true or false?* true

Die Restriktion, dass die Ameise parallel zu einer der Wände laufen muss, kann, ohne den Algorithmus aus (a) ungültig zu machen, mindestens folgendermassen abgeschwächt werden: die Ameise muss nur geradeaus in eine Richtung laufen aber nicht notwendigerweise parallel zu den Wänden.

The restriction that the ant runs parallel to one of the walls can be weakened without compromising the algorithm of (a) at least in the following way: the ant has to walk straight on in one direction but not necessarily parallel to one of the walls.

(c) Wahr oder falsch? *true or false?*

2 P

Die Restriktion, dass die Ameise Ihre Laufrichtung ständig beibehalten muss, kann, ohne den Algorithmus aus (a) ungültig zu machen, mindestens folgendermassen abgeschwächt werden: die Ameise bewegt sich weiterhin auf einem Strahl, darf aber Ihre Laufrichtung endlich oft ändern, d.h. hin und her laufen. Wir gehen davon aus, dass die Ameise nicht auf der Begrenzung des Polygons umkehrt. Begründen Sie ganz kurz.

The restriction that the ant must constantly keep its direction can be weakened without compromising the algorithm of (a) at least in the following way: the ant keeps moving on a straight line but may reverse its direction finitely often, i.e. moving forwards and backwards. We assume that the ant does not reverse on the boundary of the polygon. Justify your answer shortly.

Cancellation of crossings (in terms of mod 2)

(d) Folgende Methode soll als Teil des Point-In-Polygon Algorithmus zurückgeben, ob für ein Polygonsegment eine echte Überquerung vorliegt. Wir nehmen an, dass die Ameise in horizontaler (x-) Richtung vom Punkt (px, py) aus startet. Wir betrachten das Polygonsegment von $(startx, starty)$ nach $(endx, endy)$ und nehmen an, dass der Endpunkt eines Teilstückes des Polygons immer zugleich Anfangspunkt eines anderen Polygonstückes ist.

The following method shall as a part of the point-in-polygon algorithm return if there is a real crossing for a polygon segment. We assume that the ant starts in horizontal (x-) direction from point (px, py) . We consider a polygonsegment from $(startx, starty)$ to $(endx, endy)$ and assume that the endpoint of a polygon piece is always also the beginning point of another polygon piece.

4 P

```
bool IsRealIntersection(int px, int py,
                       int startx, int starty, int endx, int endy)
{
    if (py > starty && py < endy || py > endy && py < starty)
        return true;
    else if (py == starty)
        return starty > endy;
    else if (py == endy)
        return starty < endy;
    else
        return false;
}
```

8 Hashtabellen (12 Punkte)

Betrachten Sie folgende vereinfachte Implementati-
on einer Hashtabelle Table für Postleitzahlen.

*Consider the following oversimplified imple-
mentation of a hash table Table for postal
codes.*

```
class Table {
    int [] keys = new int [10];
    String [] values = new String [10];
    int n = 0;

    // return an index for a given key value
    int hash(int x) {
        return x % 10;
    }

    // perform a linear probing for the key, i.e. search for key
    // starting at given index applying wraparound semantics.
    int probe(int index, int key) {
        while (keys[index] != key)
            index = (index + 1) % 10;
        return index;
    }

    // compute the starting index for key. Then get next free index
    // (with keys[index] == 0) and store key and value.
    void put (int key, String value) {
        int index = probe(hash(key), 0);
        ++n;
        keys[index] = key;
        values[index] = value;
    }

    // compute the starting index i for key. Starting from this index i, find
    // index x with keys[x] == key. return value at this index.
    String get(int key) {
        int index = probe(hash(key), key);
        return values[index];
    }

    // print hash table values
    void Report() {
        for (int i = 0; i < n; ++i) {
            System.out.print(values[i] + " ");
        }
    }
}
```


- (a) Welche Ausgabe erzeugen die folgenden Codezeilen?

What is the output of the following code?

4 P

```
Table t = new Table();
t.put(8001, "Zurich");
t.put(8002, "Zurich");
t.put(3000, "Bern");
t.put(3001, "Bern");
t.put(4000, "Basel");
t.put(4001, "Basel");
t.put(1200, "Genf");
t.Report();
```

Bern Zurich Zurich Bern Basel Basel Genf

- (b) Angenommen, die Postleitzahlen wurden genauso wie in (a) in die Tabelle t eingefüllt. Wie viele Schlüsselvergleiche der Art `keys[index] != key` sind dann nötig, um via `c.get(3001)` den Ort zur Postleitzahl 3001 zu finden?

Assume that the postal codes are entered in the table t as in (a). How many key comparisons of the form `keys[index] != key` are then required to find the city to postal code 3001 via `c.get(3001)`?

4 P

3

- (c) Angenommen eine Tabelle x vom Typ `Table` wurde mit $n < 10$ Datensätzen befüllt und ein Datensatz zum Schlüssel (Postleitzahl) k sei in x enthalten. Wie viele Schlüsselvergleiche finden beim Aufruf von `x.get(k)` maximal statt?

Assume a table x of type `Table` was filled with $n < 10$ data sets and the dataset with key (postal code) k is contained in x . How many key comparisons take place upon a call of `x.get(k)` maximally?

2 P

n

Und minimal ?

and minimally ?

1

- (d) Angenommen Sie haben eine Implementation einer dynamischen Hashtabelle vorliegen. Auch die verwendete Hashfunktion sei gut gewählt. Was würden Sie machen, wenn Sie nach der Eingabe von Daten eine signifikante Verlangsamung von Anfragen (der Art von `get`) beobachten würden?

Assume you have an implementation of a dynamic hash table. Assume furthermore that the used hash-function is well chosen. What would you do, when, after input of some data you observe a significant slowdown for queries (of the form of `get`)?

2 P

Tabelle vergrößern / Füllgrad begrenzen