

Praktikum 6 Abgabe bis 24.01.2021

Name, Vorname: Özkurt. Cihat**Mat.-Nr.:** 11148632**GMID:** inf2323**Datum:** 24.01.2021**Betreuer im Praktikum:** Gross, Julian

Aufgabe 1: Programmieren in Java

Schreiben Sie eine Klasse **Konto**, die als Instanzvariable **kontostand** vom Typ **double** hat. Außerdem soll die Klasse die folgenden Methoden haben:

- **public double getKontostand():** Der Kontostand wird zurückgegeben.
- **public void einzahlen(double betrag):** Der Betrag wird auf das Konto eingezahlt.
- **public double abheben(double betrag):** Der Betrag wird vom Konto abgezogen, wenn der Kontostand das zulässt. Wenn nicht, wird eine Meldung ausgegeben, dass keine Auszahlung erfolgen kann, weil der Kontostand das nicht hergibt.

Schreiben Sie nun eine Klasse **Geldautomat**, die die Anwendung enthält. Die Anwendung soll ein Menü bereitstellen, um den Kontostand abzufragen, einen Betrag einzuzahlen und einen Betrag abzuheben.

Lösung:

<Kopieren Sie bitte Ihr Programm hier hin. >

```
package Praktikum6Aufgabe1;

public class Konto {
    double kontostand;
    public Konto(double kontostand){
        this.kontostand = kontostand;
    }
    public double getKontostand() {
        return kontostand;
    }

    public void einzahlen(double betrag){
        kontostand = kontostand + betrag;
        System.out.println(betrag + " wurde eingezahlt " + "aktuelle kontostand ist " + kontostand);
    }

    public void setKontostand(double kontostand) {
        this.kontostand = kontostand;
    }
    public double abheben(double betrag){
        if(betrag > kontostand)
            System.out.println("keine Auszahlung erfolgen kann ihr Betrag ist größer als ihr Kontostand");
        else{
            kontostand = kontostand - betrag;
            System.out.println(betrag + " wurde abgehebt " + "aktuelle kontostand ist " + kontostand);
        }
    }
}
```

```

        return kontostand;
    }

}

package Praktikum6Aufgabe1;

import java.util.Scanner;

public class Geldautomat {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Konto konto = new Konto(0);
        System.out.println("Hallo\n"
            + "drücke 1 um einzuzahlen\n"
            + "drücke 2 um Geld abzuheben\n"
            + "drücke 3 um Kontostand zu sehen\n"
            + "drücke 4 Ende des Programms\n");
        while(true){
            System.out.println("Was wollen Sie tun?");
            int n = scanner.nextInt();
            if(n==1){
                System.out.print("Betrag = ");
                double be trag = scanner.nextDouble();
                konto.einzahlen(be trag);
            }else if(n==2){
                System.out.print("Betrag = ");
                double be trag = scanner.nextDouble();
                System.out.println(konto.abheben(be trag));
            }else if(n==3)
                System.out.println("Kontostand = " + konto.getKontostand());
            else if(n==4){
                System.out.println("Ende des Programms");
                break;
            }
        }
    }
}

```

Aufgabe 2: Programmieren in Java

Ergänzen Sie bitte das folgende Programmsystem. Es handelt sich um ein **kleines Schweine-Kartoffel-Szenario**. Die folgende Situation ist gegeben:

- Das Schwein frisst Kartoffeln, indem es eine Kartoffel pflückt.
- Dadurch kann es seinen Nährwert erhöhen und erhält so die Energie sich zu bewegen.

Hinweis: Bitte dokumentieren Sie alles in javadoc!

Die folgende Klasse **Kartoffel** ist teilweise vorgegeben. Bitte ergänzen Sie die unterstrichenen Stellen:

```

public class Kartoffel {
    private int dicke;

    // Bitte den Konstruktor für eine Kartoffel ergänzen:
    public Kartoffel(int dicke) {

```

```

    }

    // Bitte die Methode pfluecken implementieren:
    // pfluecken gibt die Dicke der Kartoffel als Returnwert zurück.
    public int pfluecken() {

    }
}

```

Nun folgt die Klasse `Schwein`. Bitte ergänzen Sie die unterstrichenen Stellen.

```

public class Schwein {
    private String name;
    private int gewicht;
    private int saettigungsgrad;

    // Bitte den Konstruktor für ein Schwein ergänzen:

    /**
     * Konstruktor:
     * @param name
     * @param gewicht
     * @param saettigungsgrad
     */
    _____
    _____
    _____
}

// Bitte die Methode bewegen ergänzen:
// Das Schwein kann sich nur bewegen, wenn es entweder seinen
// Sättigungsgrad oder sein Gewicht verringert.
// Der Sättigungsgrad muss mindestens 4 Einheiten sein, damit sich
// das Schwein bewegen kann.
// Bewegt sich das Schwein, sinkt er um 4 Einheiten. Das Gewicht des
// Schweins nimmt nur ab, wenn es mehr als 10 beträgt.
// Es nimmt dann um 1 Einheit ab. Sind beide Bedingungen nicht
// erfüllt, so grunzt das Schwein: "Ich bin zu schlapp!"

public void bewegen() {
    _____
    _____
    _____
    _____
    _____
}

// Bitte die Methode fressen ergänzen:
// Es muss eine Kartoffel gepflückt werden. Die Dicke ergibt den
// Nährwert. Ist der Nährwert 0, so erfolgt eine entsprechende
// Meldung des Schweins: "Schade! Wieder nix zu fressen."
// Im anderen Fall steigt der Sättigungsgrad des Schweins um den
// Nährwert. Das Schwein kann seinen maximalen Sättigungsgrad jedoch
// nicht über 100 steigern. In diesem Fall erhöht das Schwein sein
// Gewicht und zwar nach der Formel:
// gewicht+=(saettigungsgrad-100)/2 und der Sättigungsgrad erhält den
// Maximalwert 100.

```

```

public void fressen(Kartoffel k) {

    // Der Nährwert ergibt sich aus der gepflückten Kartoffel:
    _____

    // Ist der Nährwert <= 0, so sagt das Schwein:
    // "Schade. Wieder nix zu fressen!".

    _____

    } else {

        // Andernfalls Algorithmus von oben.

        _____
        _____
        _____
        _____

    }

    // Bitte die Methode grunzen ergänzen:
    // Das Schwein sagt:
    // "Hallo, ich bin <Name des Schweins>.
    // Mein Gewicht ist: <gewicht>.
    // Mein Sättigungsgrad ist: <saettigungsgrad>."

    public void grunzen() {

        _____
        _____
        _____

    }

}

```

Hier kommt nun die Anwendung für unser Schwein-Kartoffel-Szenario (Klasse **Main**). Die Klasse **Main** besitzt nur die **main**-Methode.

```

public class Main {
    public static void main(String[] args) {
        // Erzeugen Sie ein Schwein mit dem Namen Fritz, dem Gewicht 10
        // und dem Sättigungsgrad 10.

        _____

        // Erzeugen Sie 2 Kartoffeln mit der Dicke 10 und 20.

        _____
        _____

        // Unser Schwein soll nun die folgenden Methoden ausführen:
        // grunzen - Kartoffel 1 fressen - grunzen - sich bewegen -
        // grunzen - Kartoffel 2 fressen - grunzen - sich bewegen -
        // grunzen.

        _____
        _____
        _____
        _____
        _____

    }

}

```

Lösung:

```
package Praktikum6Aufgabe2;
```

```
public class Kartoffel {
    private int dicke;
    // Bitte den Konstruktor für eine Kartoffel ergänzen:
    public Kartoffel(int dicke) {
        this.dicke = dicke;
    }

    // Bitte die Methode pfluecken implementieren:
    // pfluecken gibt die Dicke der Kartoffel als Returnwert zurück.
    public int pfluecken() {
        return dicke;
    }
}
```

```
package Praktikum6Aufgabe2;
```

```
public class Schwein {
    private String name;
    private int gewicht;
    private int saettigungsgrad;

    // Bitte den Konstruktor für ein Schwein ergänzen:
    /**
     * Konstruktor:
     * @param name
     * @param gewicht
     * @param saettigungsgrad
     */
    public Schwein(String name,int gewicht,int saettigungsgrad){
        this.name = name;
        this.gewicht = gewicht;
        this.saettigungsgrad = saettigungsgrad;
    }

    // Bitte die Methode bewegen ergänzen:
    // Das Schwein kann sich nur bewegen, wenn es entweder seinen
    // Sättigungsgrad oder sein Gewicht verringert.
    // Der Sättigungsgrad muss mindestens 4 Einheiten sein, damit sich
    // das Schwein bewegen kann.
    // Bewegt sich das Schwein, sinkt er um 4 Einheiten. Das Gewicht des
    // Schweins nimmt nur ab, wenn es mehr als 10 beträgt.
    // Es nimmt dann um 1 Einheit ab. Sind beide Bedingungen nicht
    // erfüllt, so grunzt das Schwein: "Ich bin zu schlapp!"
    public void bewegen() {
        if(saettigungsgrad >= 4 ){
            System.out.println("Schwein bewegt sich");
            saettigungsgrad = saettigungsgrad - 4;
            if(gewicht > 10){
                gewicht = gewicht - 1;
            }
        }
        else
            System.out.println("Ich bin zu schlapp!");
    }

    // Bitte die Methode fressen ergänzen:
    // Es muss eine Kartoffel gepflückt werden. Die Dicke ergibt den
    // Nährwert. Ist der Nährwert 0, so erfolgt eine entsprechende
    // Meldung des Schweins: "Schade! Wieder nix zu fressen."
    // Im anderen Fall steigt der Sättigungsgrad des Schweins um den
    // Nährwert. Das Schwein kann seinen maximalen Sättigungsgrad jedoch
    // nicht über 100 steigern. In diesem Fall erhöht das Schwein sein
    // Gewicht und zwar nach der Formel:
    // gewicht+=(saettigungsgrad-100)/2 und der Sättigungsgrad erhält den
    // Maximalwert 100.
    public void fressen(Kartoffel k) {
        int naehrwert = k.pfluecken();
        if(naehrwert == 0){
            System.out.println("Schade! Wieder nix zu fressen.");
        }
    }
}
```

```

else{
    saettigungsgrad = saettigungsgrad + naehrwert;

    if(saettigungsgrad + k.pfluecken() <= 100){
        saettigungsgrad = saettigungsgrad + k.pfluecken();
    }
    if(saettigungsgrad>100){
        gewicht += (saettigungsgrad-100)/2;
        saettigungsgrad=100;
    }
}

// Der Nährwert ergibt sich aus der gepflückten Kartoffel:
// Ist der Nährwert <= 0, so sagt das Schwein:
// "Schade. Wieder nix zu fressen!".
// Andernfalls Algorithmus von oben.

}
// Bitte die Methode grunzen ergänzen:
// Das Schwein sagt:
// "Hallo, ich bin <Name des Schweins>.
// Mein Gewicht ist: <gewicht>.
// Mein Sättigungsgrad ist: <saettigungsgrad>."
public void grunzen() {
    System.out.println("hallo,ich bin " + name);
    System.out.println("Mein Gewicht ist: " + gewicht);
    System.out.println("Mein Sättigungsgrad ist: " + saettigungsgrad + "\n");
}

}

package Praktikum6Aufgabe2;

public class Main {
    public static void main(String[] args) {
        Schwein schwein = new Schwein("Fritz",10,10);
        Kartoffel kartoffel_1 = new Kartoffel(10);
        Kartoffel kartoffel_2 = new Kartoffel(20);
        schwein.grunzen();
        schwein.fressen(kartoffel_1);
        schwein.grunzen();
        schwein.bewegen();
        schwein.grunzen();
        schwein.fressen(kartoffel_2);
        schwein.grunzen();
        schwein.bewegen();
        schwein.grunzen();

        // Erzeugen Sie ein Schwein mit dem Namen Fritz, dem Gewicht 10
        // und dem Sättigungsgrad 10.

        // Erzeugen Sie 2 Kartoffeln mit der Dicke 10 und 20.

        // Unser Schwein soll nun die folgenden Methoden ausführen:
        // grunzen – Kartoffel 1 fressen – grunzen - sich bewegen –
        // grunzen – Kartoffel 2 fressen – grunzen - sich bewegen –
        // grunzen.

    }
}

```

Aufgabe 3: Programmieren in Java

Schauen Sie sich das untenstehende Java Programm und beschreiben Sie **mündlich**, was hier passiert. **Wie sieht die Ausgabe aus? Warum?**

Ich habe mit Kommentar Block habe ich geklärt, was hier passiert

```
1 public class VectorsArray {
2     public static void main(String[]args) {
3         String[] VectorsLieblingsObst = {"Birne", "Apfel", "Kiwi"};
4
5         //Wir haben ein Array geschafft.Sein Typ ist String und 3 Elemente
        wurde drin eingestellt.
6
7         System.out.println(VectorsLieblingsObst[1]);
8
9         //Ausgabe der erste Element von unseren Array
10        int[] VectorsLieblingsZahlen = new int[7];
11
12        //Ein int Array wurde geschafft.Sein Begrenz ist 7 also 7 Elemente
        können drinn eingestellt werden
13
14        VectorsLieblingsZahlen[0] = 7;
15
16        //Erste Element ist 7
17
18        for(int i = 1; i < VectorsLieblingsZahlen.length; i++){
19            VectorsLieblingsZahlen[i] = i + 3;
20        }
21
22        //Mit For Schleife zuordnen wir array. Index startet mit 1 und
        weitergeht bis am Ende von Array. Jedes mal zuordnen wir i. index von array
        i + 3
23
24        System.out.printf("%nVectors Lieblingszahlen: ");
25        for (int each: VectorsLieblingsZahlen)
26            System.out.print(each + " ");
27
28        // Mit ForEach Schleife geben wir alle Zahlen in
        VectorsLieblingsZahlen Array aus.
29
30        int[] KohlsLieblingsZahlen = {2 , 5, 58, 4, 3, 444, 25};
31
32        //Array int KohlsLieblingsZahlen wurde geschafft und diese
        Elementen ordnen in KohlsLieblingsZahlen zu
33
34        VectorsLieblingsZahlen = KohlsLieblingsZahlen;
35
36        //Elementen von KohlsLieblingsZahlen wurde in Array von
        VectorsLieblingsZahlen Elementen zugeordnet.und nun sind beide Array
        gleich.Wenn man Elementen von VectorsLieblingsZahlen ändern wird
        KohlsLieblingsZahlen automatisch mit gemeinsam Wert geändert werden
```

```
z.b VictorsLieblingsZahlen[2] = 10;
```

Das bedeutet VictorsLieblingsZahlen[2] = 10 und KohlsLieblingsZahlen[2] = 10

```
20
21      VictorsLieblingsZahlen[1] = 100;

      //100 wurde in erste Element von VictorsLieblingsZahlen
zugeordnet Deswegen ist auch KohlsLieblingsZahlen[1] = 100
22
23      System.out.printf("%nKohls Lieblingszahlen: ");
24      for(int each: KohlsLieblingsZahlen)
25          System.out.print(each + " ");
26      }

      // Ausgabe alle Elementen von KohlsLieblingsZahlen
      Es werden alle Elementen von KohlsLieblingsZahlen geschrieben. Aber
1.index ist 100
27 }
```

Aufgabe 4: Programmieren in Java

Schreiben Sie eine **rekursive Methode** für die *Ackermann-Funktion*.

Die Ackermann-Funktion ist eine Abbildung von zwei ganzen Zahlen **n** und **m** auf eine ganze Zahl **a(n,m)**:

```
a(0,m) = m + 1
a(n,0) = a(n - 1, 1)
a(n,m) = a(n - 1, a(n, m - 1))
```

Schreiben Sie in die gleiche Klasse auch eine **main**-Methode, die zwei Zahlen von der Tastatur einliest und dann das Ackermann-Ergebnis für diese Zahlen ausgibt. Bitte testen Sie, ab wann der Berechnungsaufwand zu groß wird, so dass wir nicht auf das Ergebnis warten können.

Lösung:

<Kopieren Sie bitte Ihr Programm hier hin. >

```
package Praktikum6Aufgabe4;

import java.util.Scanner;

public class Main {

    public static int a(int n,int m){
        if(n==0)
            return m+1;
        else if(n>0 && m==0)
            return a(n-1,1);
        else if(n>0 && m>0)
            return a(n-1,a(n,m-1));
        return 0;
    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("n = ");
        int n = scanner.nextInt();
        System.out.print("m = ");
        int m = scanner.nextInt();

        System.out.println(a(n,m));
    }
}
```


