



WS 2005/2006

Prof. Dr. Bernhard Beckert
Dr. Manfred Jackel

Nachklausur

24.04.2006

Bitte in Druckschrift leserlich ausfüllen!

Name M u s t e r l ö s u n g

Vorname _____

E-Mail-Adresse _____ j k l @uni-koblenz.de

Matrikelnummer _____

Studiengang:

- Computervisualistik (Diplom)
- Informatik (Diplom)
- Informationsmanagement (BSc)
- Informationsmanagement (MSc)
- Anglistik & Medienmanagement (BSc)

Ich habe die Hauptklausur als Freiversuch mitgeschrieben. Diese Klausur soll meine Note verbessern. Die Note des Freiversuchs war _____ .

Diese Prüfungsleistung melde ich verbindlich als Freiversuch im Sinne der Prüfungsordnung an.

Auswertung:

	1	2	3	4	5	6	GESAMT
Punkte	12	8	12	12	12	16	72

Aufgabe 1 (12 Punkte)

Schreiben Sie ein Programm `public class KQuadrat`, das zu einer gegebenen positiven ganzen Zahl n die größte ganze Zahl q berechnet, so dass $q^2 \leq n$ gilt. Die Zahl n werde dem Programm als Kommandozeilenparameter übergeben, die Berechnung kann direkt in der `main`-Methode ausgeführt werden.

```
public class KQuadrat {  
  
    public static void main(String[] args) { // jetzt sind Sie dran!  
  
        int n = Integer.parseInt(args[0]);  
  
        int q=0;  
  
        while (q*q<=n) {  
            q++;  
        }  
        // Hier gilt  $q \cdot q > n$  und  $(q-1) \cdot (q-1) \leq n$   
        // Also ist  $q-1$  der gesuchte Wert;  
  
        q--;  
  
        System.out.println("Die gesuchte Zahl ist "+q);  
  
    }  
}
```

Aufgabe 2 (8 Punkte)

Gegeben sei folgende rekursive Java-Funktion f für positive Argumentwerte n:

```
public static int f(int n) {  
    if (n = 1)  
        return 0;  
    else {  
        int h = f(n/2);  
        return h+1;  
    }  
}
```

a) Welcher Wert wird k durch den Aufruf

int k=f(9); zugewiesen?

Begründen Sie Ihre Antwort, indem Sie das Ergebnis jedes Aufrufs der Funktion f protokollieren. Geben Sie das Argument des jeweiligen Aufrufs an.

Aufruf mit Parameter	Ergebnis des Aufrufes
f(9)	3
f(4)	2
f(2)	1
f(1)	0

b) Was berechnet f? Geben Sie eine mathematische Formel oder eine verbale Beschreibung der Funktion an.

f(n) = q , so dass q die größte ganze positive Zahl ist mit $2^q \leq n$.

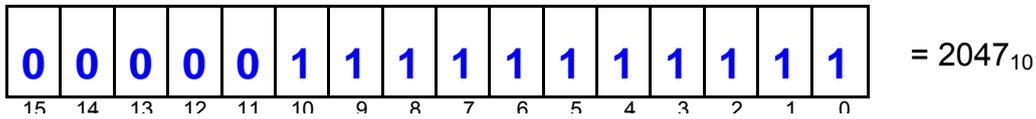
Mathematische Formel: $f(n) = \lfloor \log_2(n) \rfloor$

Aufgabe 3 (12 Punkte)

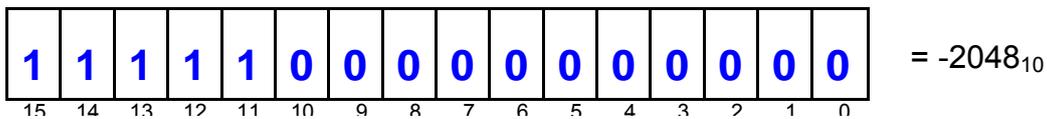
Wir verwenden im Folgenden die Binärdarstellung ganzer Zahlen im Zweierkomplement mit 16 Bit.

- Geben Sie die Binärdarstellung von 2047_{10} an!
- Geben Sie die Binärdarstellung von -2048_{10} an!
- Welche Dezimalzahl hat die Binärdarstellung 111111111100000 ?

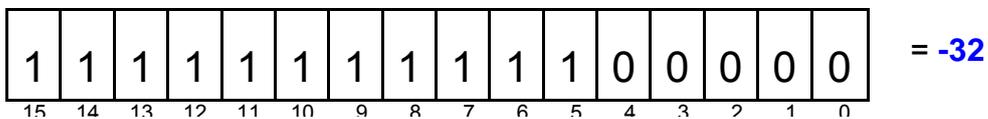
a)



b)



c)



Aufgabe 4 (12 Punkte)

- a) Formalisieren Sie die Aussagen 1-5 in Aussagenlogik, indem Sie entsprechende aussagenlogische Formeln angeben. Verwenden Sie die fett gedruckten Anfangsbuchstaben als Variablennamen (z.B. H für **Herr Meier**).

``Meiers werden uns heute abend besuchen'', kündigt Herr Müller an. ``Die ganze Familie, also Herr und Frau Meier nebst ihren drei Söhnen Tim, Kay und Uwe?'', fragte Frau Müller bestürzt. Darauf Herr Müller, der keine Gelegenheit vorbegehen lässt, seine Frau zum logischen Denken anzureizen: ``Nein, ich will es dir so erklären:

1. Wenn **Herr Meier** kommt, dann bringt er auch seine **Frau** mit.
2. Mindestens einer der beiden Söhne **Uwe** und **Kay** kommt.
3. Entweder kommt **Frau Meier** oder **Tim**.
4. Entweder kommen **Tim** und **Kay** oder beide nicht.
5. Und wenn **Uwe** kommt, dann auch **Kay** und **Herr Meier**.

Hinweis: Sie sollen nicht ausrechnen, wer kommt. Sie sollen nur die Aussagen formalisieren!

1. $H \Rightarrow F$
2. $U \vee K$
3. $F \text{ xor } T$ oder gleichwertig $(F \wedge \neg T) \vee (\neg F \wedge T)$
4. $T \Leftrightarrow K$
5. $U \Rightarrow (K \wedge H)$

- b) Überprüfen Sie mit Hilfe einer Wahrheitstabelle, ob $p \wedge (p \Rightarrow q)$ eine Tautologie ist, d.h. für alle Belegungen der Variablen wahr ist.

p	q	$p \Rightarrow q$	$p \wedge (p \Rightarrow q)$
<i>false</i>	<i>false</i>	<i>true</i>	<i>false</i>
<i>false</i>	<i>true</i>	<i>true</i>	<i>false</i>
<i>true</i>	<i>false</i>	<i>false</i>	<i>false</i>
<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>

- $p \wedge (p \Rightarrow q)$ ist allgemeingültig
 $p \wedge (p \Rightarrow q)$ ist nicht allgemeingültig

Aufgabe 5 (12 Punkte)

- Implementieren Sie eine Klasse `SamstagsZiehung`, die das Ergebnis einer Samstagsauslosung im Deutschen Lottoblock darstellen kann. Bei der Auslosung werden zunächst 6 Zahlen (aus der Ausgangsmenge 1..49) gezogen, dann aus der Restmenge eine Zusatzzahl. Weiter wird eine Superzahl zwischen 0 und 9 gezogen. Die 6 Gewinnzahlen sollen in Ziehungsreihenfolge in ein Attributfeld `zahlen`, die Zusatzzahl und die Superzahl in Attributen `zusatzZahl` bzw. `superZahl` gespeichert werden können.
- Implementieren Sie einen Konstruktor der Klasse `SamstagsZiehung`, der als Parameter die 6 Gewinnzahlen sowie die Zusatzzahl und Superzahl besitzt und die Attribute der Klasse initialisiert.
- Implementieren Sie unter Verwendung der Klasse `SamstagsZiehung` eine Klasse `JahresZiehung`, die alle 52 Ziehungen eines Jahres darstellen kann.
- Im Objekt `lotto2005` der Klasse `JahresZiehung` habe die Lottogesellschaft alle Ziehungen des Jahres 2005 abgespeichert. Vervollständigen Sie die Anweisung

```
int zerofirst = ... ;
```

so, dass in `zerofirst` die erste im Jahr 2005 gezogene Gewinnzahl aus dem Objekt `lotto2005` gespeichert wird.

a) b)

```
public class SamstagZiehung {
    int[] zahlen=new int[6];
    int zusatzZahl;
    int superZahl;
```

```
    public SamstagZiehung(int n0, int n1, int n2, int n3, int n4, int
n5, int z, int s) {
        zahlen[0]=n0;
        zahlen[1]=n1;
        zahlen[2]=n2;
        zahlen[3]=n3;
        zahlen[4]=n4;
        zahlen[5]=n5;
        zusatzZahl =z;
        superZahl =s;
    }
}
```

c)

```
public class JahresZiehung {
    SamstagZiehung[] woche = new SamstagZiehung[52];
}
```

d)

```
int zerofirst = lotto2005.woche[0].zahlen[0];
```

Aufgabe 6 (16 Punkte)

Diese Aufgabe umfasst 4 Multiple-Choice Cluster mit je 4 Ankreuzfragen. Für jedes Cluster gilt: wenn alle 4 Kreuze an der richtigen Stelle stehen, gibt es 4 Punkte für das Cluster. Ein falsches Kreuz gibt einen Punkt Abzug. Wer 2 richtige und 2 falsche Kreuzchen in einem Cluster macht, erhält $1+1-1-1=0$ Punkte. Zum Trost: es gibt keine negativen Gesamtpunktzahlen, jedes Cluster bringt 0 bis 4 Punkte.

		Ja	Nein	Frage
a)				Grundlagen
	1.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Unter Komplexität versteht man den Aufwand gemessen an der Anzahl der Eingabewerte.
	2.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Jedes terminierende Programm repräsentiert einen Algorithmus.
	3.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Rekursive Programme sind immer effizienter als iterative.
	4.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Für jedes berechenbare Problem kann man ein Java-Programm schreiben.
b)				Java
	1.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<code>int real;</code> ist korrekter Java-Code.
	2.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Jede direkt ausführbare Java-Klasse muss eine Methode <code>public void Main(String args)</code> besitzen.
	3.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Jede for-Schleife kann durch eine einzelne do-while-Schleife ersetzt werden, ohne die Anweisungen im Schleifeninneren zu ändern.
	4.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Eine Java-Klasse darf höchstens einen Konstruktor haben.
c)				Programmieren
	1.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Java kennt keine Rekursion.
	2.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Ein Programm ist genau dann partiell korrekt, wenn es für mindestens eine Eingabe terminiert und korrekte Ergebnisse berechnet.
	3.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Die Datenstruktur „Stapel“ (stack) verwendet das Last-in-first-out-Prinzip
	4.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Ein zusätzliches Element am Anfang einzusortieren, ist bei verzeigerten Listen einfacher als bei Arrays.
d)				Logik
	1.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	$(a \vee b) \wedge \neg(c \vee d)$ ist in konjunktiver Normalform
	2.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Eine Formel A ist allgemeingültig, wenn gilt: $A \equiv true$
	3.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	$\neg a \Rightarrow \neg a$ ist allgemeingültig
	4.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Für jede erfüllbare aussagenlogische Formel A gilt: $A \wedge \neg A \equiv false$