

# **ABITURPRÜFUNG 2008**

## **LEISTUNGSFACH**

### **INFORMATIK**

#### **(HAUPTTERMIN)**

Arbeitszeit: 270 Minuten

Hilfsmittel: Wörterbuch zur deutschen Rechtschreibung  
Taschenrechner (nicht programmierbar, nicht grafikfähig)  
(Schüler, die einen CAS-Taschencomputer im Unterricht benutzen, dürfen diesen verwenden.)  
Tafelwerk  
PC mit Prolog-System und  
Oberon-, Pascal- oder Java-System  
Zufallszahlengenerator

Lösen Sie die Aufgaben 1 und 2 und wählen Sie von den Aufgaben 3.1 und 3.2 eine Aufgabe zur Bearbeitung aus.

Rechts unten neben jeder Teilaufgabe steht die für diese Teilaufgabe maximal erreichbare Anzahl von Bewertungseinheiten (BE).

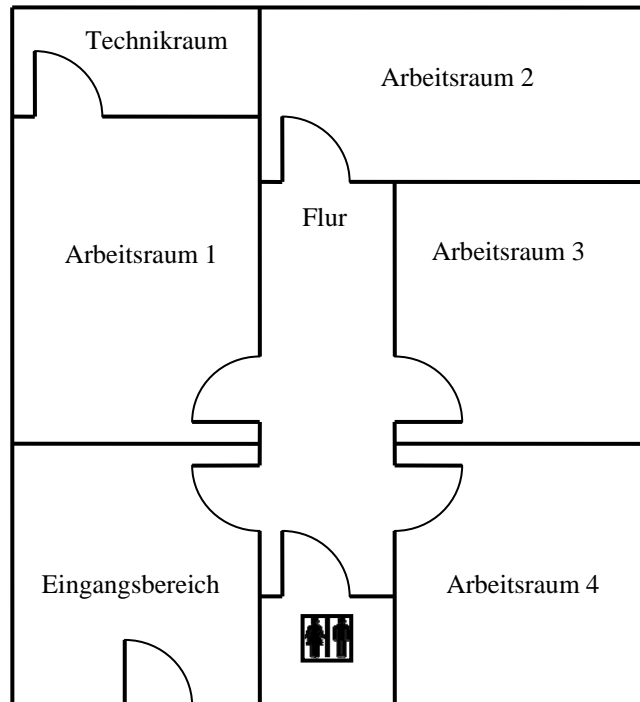
Der Prüfungsteilnehmer sichert bei der praktischen Arbeit am PC mindestens alle 10 Minuten die von ihm erarbeiteten Quelltexte. Er hat die von ihm erarbeiteten Programme und Module im Quelltext zu kommentieren. Die Quelltexte sind zusammen mit der Abiturarbeit abzugeben.

**ÖFFNUNG AM 30. APRIL 2008**

## Aufgabe 1

1.1. In der folgenden Abbildung ist der Grundriss einer Kanzlei dargestellt.

Im Eingangsbereich befinden sich ein Computer und ein Drucker, der keinen Anschluss an das Computernetz besitzt. Im Arbeitsraum 1 sind zwei Computer, im Arbeitsraum 3 sind zwei Computer, im Arbeitsraum 2 ist ein Computer und im Arbeitsraum 4 ist ein Computer aufgestellt worden. Alle sieben Computer verfügen über Netzanschlüsse. Im Technikraum befindet sich ein Drucker mit Netzanschluss. Außerdem kann im Technikraum weitere Computertechnik aufgestellt werden. Zusätzlich benötigte Computertechnik und technisches Zubehör sowie Software kann angeschafft werden.



Fertigen Sie eine Konzeption zur Vernetzung der Hardware und Installation der Software an. Beachten Sie dabei, dass jeder Angestellte der Kanzlei in seinem Sachgebiet vertrauliche Daten bearbeiten und verwalten muss. Begründen Sie die in der Konzeption getroffenen Entscheidungen.

Hinweis: Sie können das beiliegende Arbeitsblatt verwenden.

Die Kanzlei verfügt über einen Internetzugang. Von jedem Computer aus soll die Nutzung des Internets möglich sein.

Erweitern Sie Ihre Konzeption so, dass die Sicherheit im Zusammenhang mit der Nutzung des Internets in der Kanzlei gewährleistet ist. Begründen Sie Ihre Entscheidungen.

9 BE

1.2. Erläutern Sie am Beispiel des Dienstes World Wide Web (www), wie im Internet die Kommunikation und der dafür notwendige Datenaustausch zwischen der Anbieter- und der Benutzerseite geregelt werden.

6 BE

## Aufgabe 2

Gegeben ist eine Grammatik der Sprache der gebrochenen Dualzahlen:

```

gebrocheneDualzahl = "0".
gebrocheneDualzahl = "1".
gebrocheneDualzahl = "0" ganzerAnteil.
gebrocheneDualzahl = "1" ganzerAnteil.
ganzerAnteil       = "0" ganzerAnteil.
ganzerAnteil       = "1" ganzerAnteil.
ganzerAnteil       = "." gebrochenerAnteil.
gebrochenerAnteil = "0" gebrochenerAnteil.
gebrochenerAnteil = "1" gebrochenerAnteil.
gebrochenerAnteil = "0".
gebrochenerAnteil = "1".
  
```

Das Startsymbol ist gebrocheneDualzahl.

Geben Sie mithilfe dieser Grammatik drei verschiedene syntaktisch korrekte gebrochene Dualzahlen an.

Wenden Sie die folgenden Aussagen Niklaus Wirths auf die gegebene Grammatik an:

„In jeder Sprache gibt es Folgen von ... Symbolen, die als korrekt oder wohlgeformt, andere die als falsch oder missgebildet gelten. In erster Linie ist es die Grammatik ..., die bestimmt, zu welcher Kategorie eine Symbolfolge gehört.“

Stellen Sie die gegebene Grammatik mithilfe der folgenden Operatoren in der erweiterten Backus-Naur-Form (EBNF) dar:

Operation	Beispiel	Bedeutung
Sequenz	A B	B folgt unmittelbar auf A
Auswahl	A   B	entweder A oder B
Option	[A]	A kann einmal oder gar nicht vorkommen
Wiederholung	{A}	A kann 0-mal oder beliebig oft vorkommen

Entwerfen und implementieren Sie ein Programm in Prolog. Das Programm prüft, ob die in einer Anfrage gegebene gebrochene Dualzahl syntaktisch korrekt ist. Das Prüfen hat auf der Grundlage der gegebenen Grammatik zu erfolgen.

Hinweis:

Sie dürfen gebrochene Dualzahlen auch mithilfe von Prolog-Listen darstellen.

Geben Sie zum Programm eine Anfrage an.

Entwerfen und implementieren Sie ein zweites Programm in Prolog. Das Programm erzeugt und gibt alle gebrochenen Dualzahlen aus, deren Anzahl von Ziffern vor und deren Anzahl von Ziffern nach dem Punkt in der Anfrage vorgegeben wird.

Hinweis:

Sie dürfen gebrochene Dualzahlen auch mithilfe von Prolog-Listen darstellen.

Geben Sie zum zweiten Programm eine Anfrage an.

15 BE
-------

### Aufgabe 3.1

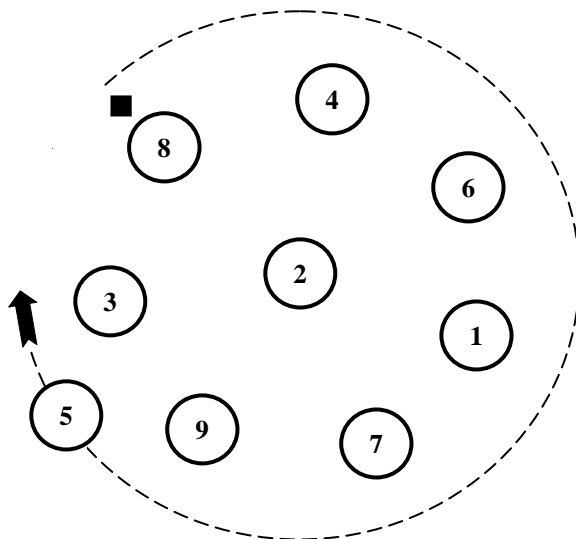
Eine Ringliste ist eine einfach verkettete Liste, bei der das letzte Listenelement auf das erste Listenelement verweist:

- Erklären Sie den Begriff einfach verkettete Liste.
- Erläutern Sie die Operation Einfügen eines Elements in eine Ringliste.

Ein Kinderspiel läuft in Runden ab und hat folgende Regeln:

In jeder Spielrunde stehen Kinder im Kreis um ein anderes Kind, das im Zentrum des Kreises steht, herum. Sie blicken auf das Kind im Kreis. Ein weiteres Kind steht außerhalb des Kreises. Dieses Kind hat ein Tuch in der Hand und ist der Plumpsack. Der Plumpsack läuft im Uhrzeigersinn außen um den Kreis herum. Hinter einem Kind lässt er das Tuch fallen. Danach können zwei Ereignisse eintreten:

1. Das Kind, hinter dem das Tuch liegt, bemerkt das Tuch. Es muss versuchen, den Plumpsack zu fangen. Dazu hebt das Kind das Tuch auf und läuft im Uhrzeigersinn hinter dem Plumpsack her.
  - Fängt das Kind den Plumpsack, bekommt der Plumpsack das Tuch wieder. Das Kind kehrt an seinen Platz zurück. Die nächste Spielrunde folgt.
  - Fängt das Kind den Plumpsack nicht, stellt sich der Plumpsack auf den Platz, auf dem das Kind stand. Das Kind bleibt außerhalb des Kreises stehen, behält das Tuch und ist der neue Plumpsack. Die nächste Spielrunde folgt.
2. Das Kind, hinter dem das Tuch liegt, bemerkt das Tuch nicht und bleibt stehen. Erreicht der Plumpsack den Platz, auf dem das Kind steht, muss sich das Kind in den Kreis stellen. Der Plumpsack stellt sich auf den Platz des Kindes. Das Kind, das bisher in der Mitte des Kreises stand, bekommt das Tuch und ist der neue Plumpsack. Die nächste Spielrunde folgt.



Beispiel:

In der Abbildung ist die Anordnung einer Spielrunde dargestellt. Neun Kinder 1 bis 9 nehmen am Spiel teil. Sieben Kinder stehen im Kreis um das Kind 2 herum. Das Kind 5 ist der Plumpsack. Hinter dem Kind 8 liegt das Tuch.

Entwerfen und implementieren Sie ein Programm, welches das Kinderspiel simuliert.

Beachten Sie folgende Festlegungen:

- Das Programm importiert und verwendet ein Modul, das für den Kinderkreis die Datenstruktur Ringliste bereitstellt:
  - Geben Sie eine Spezifikation der Ringliste an.
  - Implementieren Sie die Ringliste in einem Modul.
- Das Programm liest die Anzahl der Kinder ein.
- Das Programm legt zu Beginn fest:
  - die Nummer des Kindes, das als erstes der Plumpsack ist
  - die Nummer des Kindes, das als erstes im Zentrum des Kreises steht
  - die Nummern der Kinder, die sich im Kreis aufstellen
- Das Programm soll für jede Spielrunde Folgendes ausgeben:
  - im Uhrzeigersinn die Nummern der Kinder, die den Kreis bilden
  - die Nummer des Kindes, das im Kreis steht
  - die Nummer des Kindes, das der Plumpsack ist
  - die Nummer des Kindes, hinter dem der Plumpsack das Tuch fallen lässt
- Mit Hilfe des Zufallsgenerators wird in jeder Spielrunde festgelegt:
  - die Nummer des Kindes, hinter dem der Plumpsack das Tuch fallen lässt
  - ob der Plumpsack gefangen wird

Dokumentieren Sie einen Testlauf des Programms.

Lassen Sie dazu das Programm zehn Spielrunden simulieren.

Erläutern Sie, welche Methoden der Software-Entwicklung Sie bei der Erstellung des Programms verwendet haben.

30 BE
-------

### Aufgabe 3.2

Gegeben sind eine Turing-Maschine A und eine Turing-Maschine B. Die Turing-Maschinen besitzen das Alphabet  $\{*, 0, 1\}$  und die Menge der Bewegungen des Lese- und Schreibkopfs  $\{\text{links, keine, rechts}\}$ . Am Anfang steht der Lese- und Schreibkopf auf dem ersten Zeichen des Eingabeworts. Ein Eingabewort enthält endlich viele Zeichen "1" oder "0" und beginnt mit "1". Links und rechts vom Eingabewort stehen nur Zeichen "\*".

Die Turing-Maschine A hat die Menge der Zustände  $\{z_0, z_1, z_2, z_3\}$ , den Anfangszustand  $z_0$ , den Endzustand  $z_3$  und die folgende tabellarisch gegebene Überföhrungsfunktion f:

	$z_0$	$z_1$	$z_2$	$z_3$
*	* links $z_1$	* keine $z_3$	* keine $z_3$	* keine $z_3$
0	0 rechts $z_0$	0 links $z_2$	* links $z_2$	0 keine $z_3$
1	1 rechts $z_0$	1 links $z_2$	* links $z_2$	1 keine $z_3$

Fertigen Sie zum Eingabewort 1101 und zum Eingabewort 10010 je ein Ablaufprotokoll der Turing-Maschine A an. Geben Sie jeweils das Ausgabewort an.

Geben Sie an, was die Turing-Maschine A leistet.

Die Turing-Maschine B soll jede 0 im Eingabewort durch eine 1 und jede 1 im Eingabewort durch eine 0 ersetzen.

Beispiel:

Eingabewort	Ausgabewort
100110000111	011001111000

Geben Sie die Menge der Zustände, den Endzustand und die Überföhrungsfunktion g der Turing-Maschine B an.

Fertigen Sie zum Eingabewort 1001 ein Ablaufprotokoll der Turing-Maschine B an. Geben Sie das Ausgabewort an.

Entwerfen und implementieren Sie ein Programm, das die Arbeitsweise der Turing-Maschine A oder B simuliert.

Dokumentieren Sie einen Testlauf des Programms.

(Fortsetzung der Aufgabe auf Seite 8)

Der 1903 geborene Mathematiker und Mitbegründer der theoretischen Informatik Alonzo Church stellte die folgende These auf:

„Alle intuitiv vom Menschen lösbaren Algorithmen kann auch eine Turing-Maschine lösen.“

Setzen Sie sich mit dieser These auseinander.

30 BE
-------