

Prüfungsklausur 31231 – WS 13/14

Prof. Dr. J. Keller

01.03.2014

1 Aufbau und Funktion eines Personal Computers

- a) Nennen Sie sechs Komponenten, die auf einem Mainboard enthalten sind (3 P.).

- b) Was versteht man unter einem Chipsatz? Nennen Sie die wichtigsten Bausteine des Chipsatzes und erläutern Sie die Aufgaben der Bausteine in wenigen Sätzen (6 P.).

- c) Welche wesentlichen Aufgaben hat das BIOS (3 P.)?

Lösungsvorschläge

Zu a)

Prozessor-Sockel

RAM-Steckplätze

PCIe-Steckplätze

Festplattenanschlüsse (z.B.: sATA)

North Bridge

South Bridge

Audio-Anschluß

Zu b)

Der Chipsatz ist eine Ansammlung von Chips die benötigt werden, um den Prozessor mit dem Speichersystem und Ein-/Ausgabebussen zu koppeln. (Also im eigentlichen Sinne die Steuerung des Systems zu übernehmen). Er besteht häufig aus ein bis drei Bausteinen. Zu den wichtigsten Bausteinen zählen die North Bridge und die South Bridge.

Die North Bridge verbindet den Prozessor mit allen Komponenten, die einen möglichst schnellen Datentransfer benötigen. Das sind insbesondere der Hauptspeicher und die Grafikeinheit. Da die North Bridge sich insbesondere um die Steuerung der Zugriffe auf den angeschlossenen Hauptspeicher kümmern muss, wird sie auch als Speicher-Controller-Hub bezeichnet.

Die South Bridge verbindet den Prozessor oder Hauptspeicher mit einer Reihe von integrierten oder extern hinzugefügten Controllern, die insbesondere zur Steuerung von Massenspeicher- oder Ein-/Ausgabegeräten dienen.

Zu c)

Zu den wesentlichen Aufgaben des BIOS zählen:

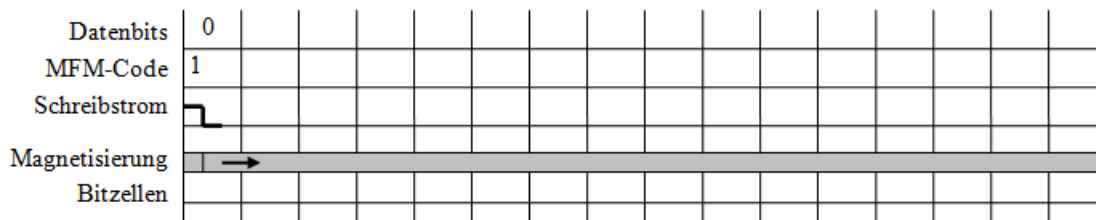
- Ausführung eines PC-Selbsttests,
- Konfiguration der auf der Hauptplatine oder in den Steckplätzen eingesetzten Komponenten,
- Ausführung eines Setup-Programms, durch das der Anwender BIOS-Daten ändern und Einfluss auf den Systembetrieb nehmen kann.

2 Codierungsverfahren

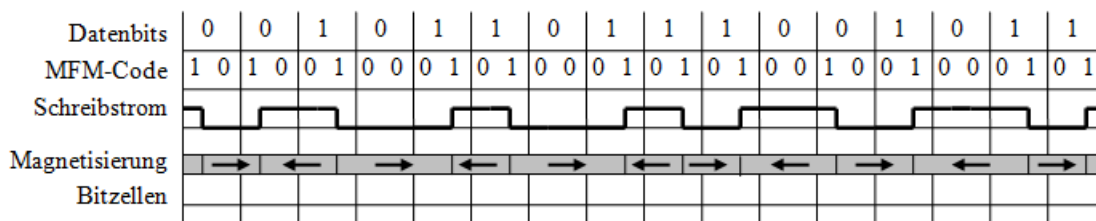
Die Datenbit-Folge $D = 0010\ 1101\ 1100\ 1011$ soll auf einer Festplatte mit dem MFM-Verfahren aufgezeichnet werden. Tragen Sie dazu in das folgende Bild die aufgezeichneten Datenbits, die resultierende MFM-Codierung, den Verlauf des Schreibstroms sowie die Magnetisierungsrichtung der Festplatte ein. Als Rastergröße in waagerechter Richtung ist darin die Länge einer Bitzelle vorgegeben, d.h. der kleinste Abschnitt gleichgerichteter Magnetisierung. Ordnen Sie die Daten- und Code-Bits jeweils über den Bitzellen an, in denen sie abgespeichert werden. (6 P.)

Hinweis: Die folgende Tabelle zeigt die Zuordnung der Datenbits bei der MFM-Codierung:

Datenbit		Speichercode
D_{n-1}	D_n	
0	0	10
1	0	00
0	1	01
1	1	01



Lösungsvorschläge



3 Speichermedien und Peripheriegeräte

Welche der folgenden Aussagen treffen zu? (6 P.)

trifft zu: trifft nicht zu:

Auf einer CD-ROM sind die Informationen in der Speicherschicht durch Erhöhungen (Lands) und Vertiefungen (Pits) codiert.

Der Break Code ist ein Scancode, der beim Drücken einer Taste ausgegeben wird.

Um die Eigenschaften einer Datei bereits am Namen erkennbar zu machen, erhalten gleichartige Dateien alle eine gleichlautende Namenserweiterung, auch Präfix genannt.

Bei modernen optischen Mäusen handelt es sich um miniaturisierte Digitalkameras, die auch als IntelliEye-Sensoren bezeichnet werden.

Der Laserstrahl eines Laserdruckers wird durch einen rotierenden Spiegel so umgelenkt, dass jeweils eine Druckzeile abgedeckt werden kann.

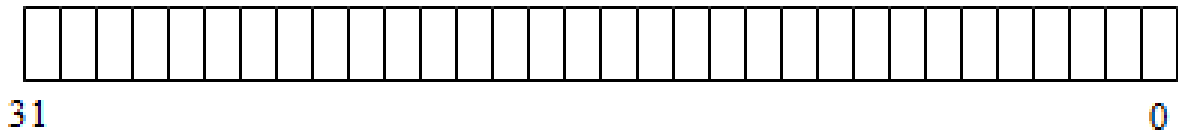
Bei einer herkömmlichen Tastatur können aus dem Tastaturlayout die Spalten und Zeilennummern für die einzelnen Tasten eindeutig zugeordnet werden.

Lösungsvorschläge

Die erste, vierte und fünfte Aussage treffen zu.

4 Datenformate

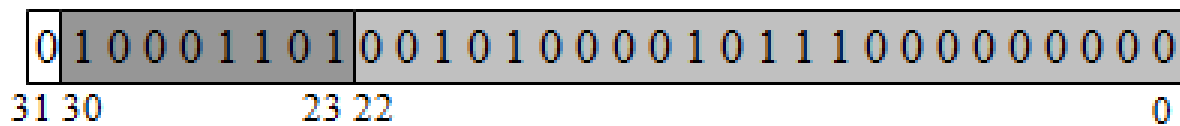
Stellen Sie die Dezimalzahl $Z_1 = 18967$ im 32-bit-Format des IEEE-754-Standards in binärer Form dar. Kennzeichnen Sie die unterscheidbaren Bitfelder (Vorzeichen, biased Exponent, Mantisse). (4 P.)



Lösungsvorschläge

$$Z_1 = 18967 = 2^{14} + 2^{11} + 2^9 + 2^4 + 2^2 + 2^1 + 2^0 = 18967 = (-1)^0 \cdot 2^{141-127} \cdot (1.001010000101110 \dots 0)_2$$

Daraus ergibt sich:



5 Organisation von Cache-Speichern

- a) Gegeben sei ein Cache C1 mit voller Zuordnungsfreiheit (vollassoziativer Cache). Der Cache umfasst 256 Blöcke von je 16 Bytes. Der Speicher ist byte-weise adressierbar, Adressen sind 16 Bit lang (Bit 15 bis Bit 0).

Geben Sie an, wieviele Bits der Adresse jeweils zu Tag, Index, und Wortadresse gehören. (3 P.)

Tag:

Index:

Wortadresse:

- b) Ändert sich die Anzahl der Bits der Wortadresse aus Teil a), wenn es sich um einen Cache mit direkter Zuordnung statt voller Zuordnungsfreiheit handelt? (1 P.)

Ja	Nein
<input type="checkbox"/>	<input type="checkbox"/>

- c) Gegeben sei ein Cache C2 mit 1024 Blöcken von je 1 Byte, mit direkter Zuordnung (direct mapped). Der Speicher ist byte-weise adressierbar, Adressen sind 16 Bit lang (Bit 15 bis Bit 0). Zu Beginn sei der Cache leer. Nun werden 100 Lesezugriffe auf verschiedene Adressen durchgeführt. Danach werden die 100 Lesezugriffe in gleicher Reihenfolge mit den gleichen Adressen wie vorher wiederholt. Geben Sie Anzahl der Misses bei den ersten 100 Lesezugriffen, sowie die minimal mögliche und maximal mögliche Anzahl von Misses bei den zweiten 100 Lesezugriffen an. (3 P.)

Anzahl Misses bei Zugriffen 1-100:

minimale Anzahl Misses bei Zugriffen 101-200:

maximale Anzahl Misses bei Zugriffen 101-200:

- d) Welche der 3 Zahlen aus Teil c) ändert sich, wenn es sich um einen Cache mit voller Zuordnungsfreiheit (voll-assoziativer Cache) handelt? (1 P.)

- e) Gegeben sei ein seitenweise orientierter Hauptspeicher mit einer virtuellen Speicherverwaltung. Geben Sie 3 verschiedene Verdrängungsstrategien für das Einlagern von Seiten nach einem Seitenfehler an. Geben Sie hierbei sowohl Akronyme als auch deren Langfassungen an, und erläutern Sie jede Strategie mit einem Satz. Welche der 3 Strategien kann an Beladys Anomalie leiden? Welche der 3 Strategien benötigt Kenntnisse über zukünftige Zugriffe? (4 P.)

Lösungsvorschläge

Zu a) Die Wortadresse umfasst 4 Bit, da jeder Block aus 16 Bytes besteht und der Speicher byte-weise adressierbar ist. Der Index umfasst 0 Bit, da es sich um einen Cache mit voller Zuordnungsfreiheit handelt. Damit umfasst der Tag 12 Bit, da die Adresse 16 Bit lang ist, und nur 4 Bit zu den anderen Teilen gehören.

Zu b) Nein, die Anzahl der Bits in der Wortadresse ist nur von der Anzahl der Bytes eines Blocks abhängig.

Zu c) Die ersten 100 Zugriffe sind alle Misses, da auf jede der Adressen erstmalig zugegriffen wird. Im günstigsten Fall haben alle Adressen unterschiedliche Indexwerte, damit gibt es bei den nächsten 100 Zugriffen nur Hits, also 0 Misses. Im schlechtesten Fall haben alle Adressen den gleichen Indexwert, damit verdrängen sich die Einträge gegenseitig, und es gibt auch bei den nächsten 100 Zugriffen 100 Misses.

Zu d) Es ändert sich lediglich die maximale Anzahl von Misses bei den zweiten 100 Zugriffen, da bei einem vollassoziativen Cache der Index keine Rolle spielt, und somit Adressen solange in verschiedenen Cache-Einträgen gespeichert werden können, wie freie Einträge vorhanden sind. Da es 1024 Einträge gibt, kann keine Verdrängung stattgefunden haben, und die zweiten 100 Zugriffe sind alle Hits.

Zu e) LRU=least-recently used

Es wird die Seite verdrängt, deren letzter Zugriff am längsten her ist.

FIFO=first in first out

Es wird die Seite verdrängt, die als erste eingelagert worden ist.

OPT=optimale Strategie

Es wird die Seite verdrängt, deren nächste Nutzung am weitesten in der Zukunft liegt.

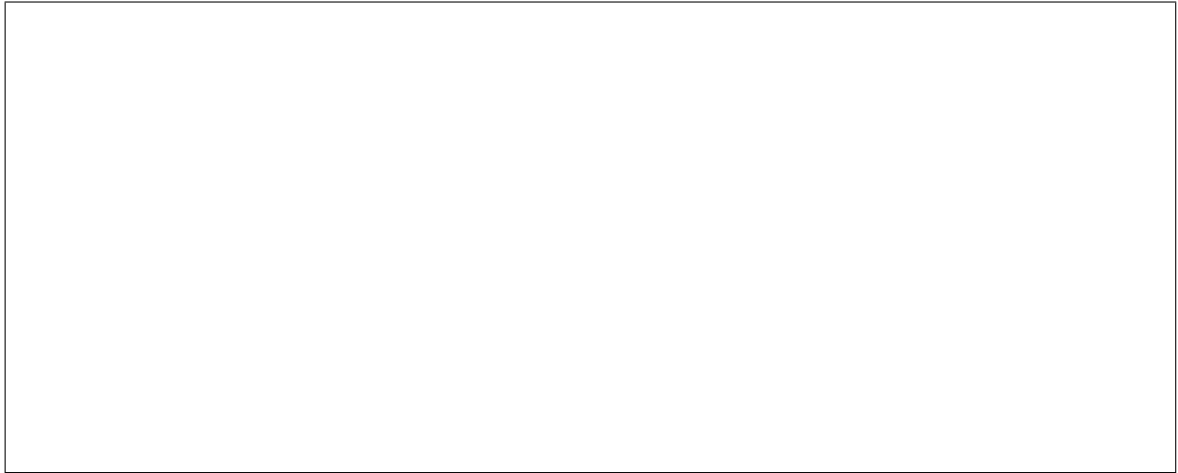
FIFO kann an Beladys Anomalie leiden, OPT benötigt Kenntnisse über zukünftige Zugriffe.

6 Endliche Automaten

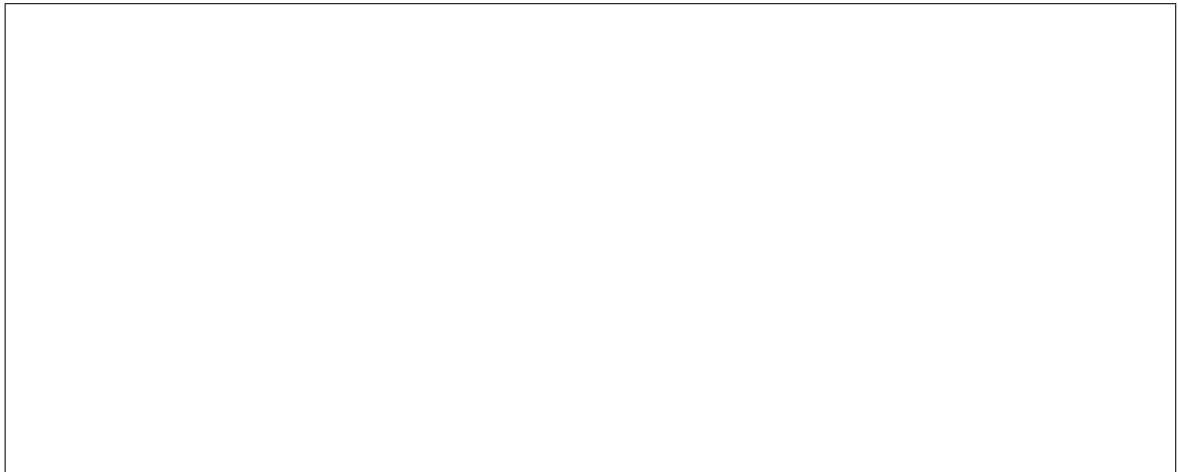
- a) Gegeben sei das Alphabet $\{a, b\}$. Geben Sie einen regulären Ausdruck an für die Sprache aus allen Zeichenreihen, die entweder nur aus a 's (mindestens 3) bestehen, oder die aus einer Folge von abwechselnd a 's und b 's bestehen, die mit einem a beginnt, aber mit einem a oder b enden darf. (2 P.)

- b) Gegeben sei das Alphabet $\{a, b, c\}$ sowie der reguläre Ausdruck $((a + b)(a + c))^*$. Entwerfen Sie einen endlichen Automaten, der die gleiche Sprache akzeptiert. Der Automat darf höchstens eine ε -Transition enthalten, und falls ein Wort nicht akzeptiert wird, soll sich der Automat am Ende in einem speziellen Zustand s_F (F für Fehler) befinden. (4 P.)

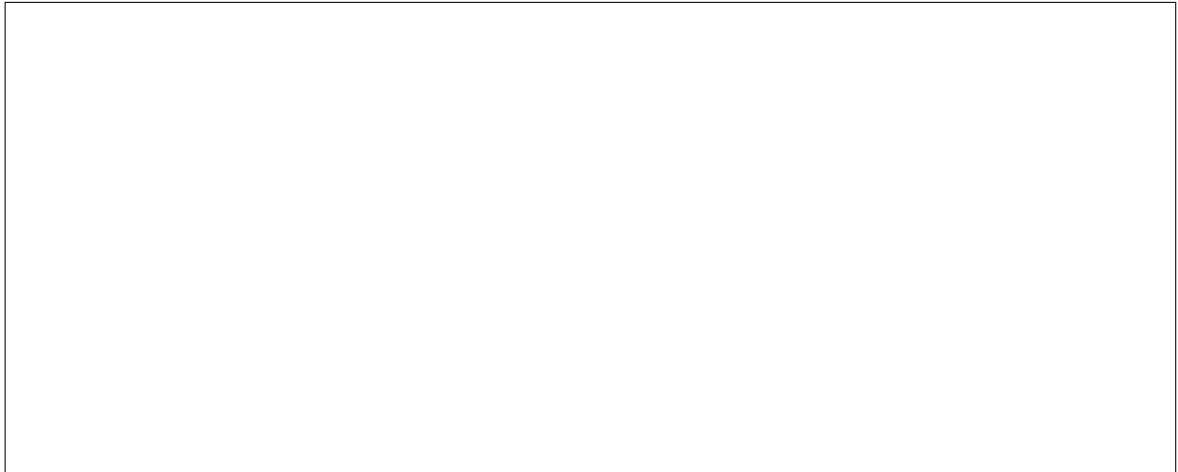
- c) Definieren Sie die Klasse \mathcal{P} der effizient lösbaren Probleme. (2 P.)



- d) Gegeben sei ein Problem aus \mathcal{NP} der Größe n und ein Algorithmus, der zur Lösung 10^n Operationen braucht. Berechnen Sie, bis zu welcher Größe n dieser Algorithmus Lösungen innerhalb von 3 Jahren findet. Nehmen Sie dabei vereinfachend an, dass eine Operation 10^{-8} Sekunden braucht, ein Tag 10^5 Sekunden hat, und 3 Jahre 10^3 Tage. (2 P.)



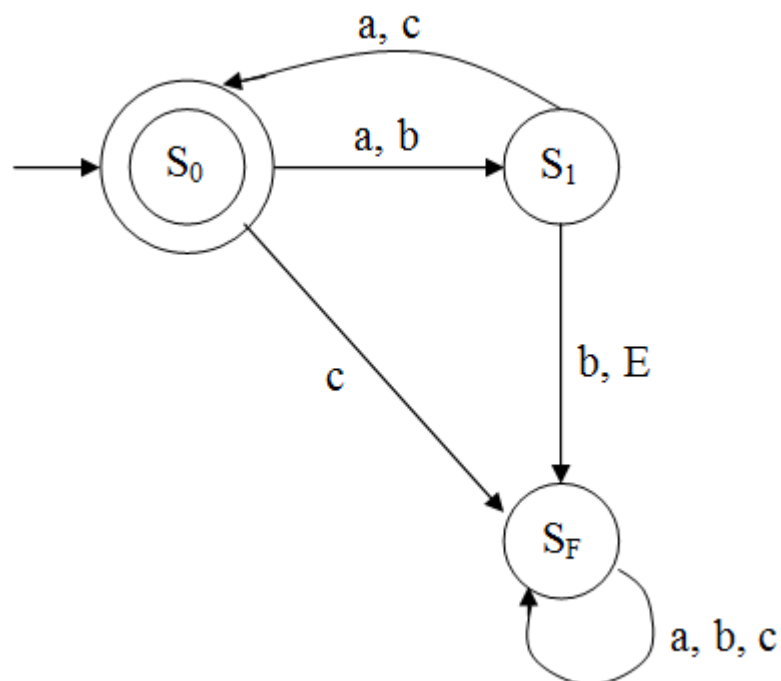
- e) Definieren Sie Approximationsalgorithmen und Heuristiken. Ist der First-Fit-Decreasing-Algorithmus für das Bin-Packing Problem ein Approximationsalgorithmus oder eine Heuristik? (3 P.)



Lösungsvorschläge

Zu a) $aaaa^* + (ab)^*a + (ab)^*ab$

Zu b)



Zu c) Die Klasse \mathcal{P} der effizient lösbaren Probleme ist die Menge aller Probleme, so dass es für jedes Problem aus \mathcal{P} einen Algorithmus gibt, der zur Lösung des betreffenden Problems der Größe n nicht mehr als $O(n^k)$ viele Schritte benötigt, wobei k eine vom Problem abhängige, feste natürliche Zahl ist.

Zu d) In 3 Jahren kann der Algorithmus $10^3 \cdot 10^5 \cdot 10^8 = 10^{16}$ Operationen ausführen, also ein Problem der Größe $n = 16$ lösen.

Zu e) Algorithmen, die polynomielle Zeit benötigen, und bei denen man beweisen kann, dass sie nie mehr als ein gewisses Quantum von der optimalen Lösung abweichen, nennt man Approximationsalgorithmen. Algorithmen, bei denen man einen solchen Beweis nicht führen kann, nennt man Heuristiken. Der First-Fit-Decreasing-Algorithmus ist eine Heuristik.