

Klausur Grundlagen der Informatik

- Bearbeitungszeit: **120 Minuten**
- es sind **keine** Unterlagen (Bücher, Vorlesungsmitschriften, Übungen), Taschenrechner, Mobiltelefone (ausschalten und in die Tasche legen) sowie Computer jeglicher Art erlaubt
- legen Sie beschriebene Lösungsblätter abgedeckt ab
- Zwischenschritte für die Berechnungen/Umstellungen sind mit anzugeben
- die Klausur ist **mit** Aufgabenzetteln abzugeben

Aufgabe 0

(1 Punkt(e))

Schreiben Sie Ihre **Matrikelnummer und Studiengang** auf jedes Blatt das in die Bewertung einbezogen werden soll! Ergänzen Sie diese Angaben auch auf der letzten Seite dieser Aufgabenzettel.

Aufgabe 1 - Zahlensysteme

(12 Punkt(e))

- 1.1 (1 Punkt) Was ist 2^{12} in dezimaler Schreibweise?
- 1.2 (4 Punkte) Addieren Sie die folgenden Zahlen schriftlich binär $0x6E$, $0b01001111$, $0d24$. Geben Sie das Ergebnis in hexadezimaler Form an.
- 1.3 (2 Punkte) Multiplizieren Sie die Dualzahlen $0b10011$ und $0b1010$ schriftlich, d.h. mit Lösungsweg und im Binärsystem.
- 1.4 (5 Punkte) Geben Sie für die Dezimalzahlen 0 und -15 alle zugehörigen 8-Bit-Darstellungen im Einer- und im Zweierkomplement an.

Aufgabe 2 - C Grundlagen

(16 Punkt(e))

- 2.1 (2 Punkte) Beschreiben Sie kurz, welche Aufgaben das folgende Code-Fragment erfüllt.

```
1  int a = 1;  int b = 1;  int c;
2  for(int idx = 0; idx < 10; idx++){
3      c = a+b;
4      a = b;
5      b = c;
6      printf("%d\n", c);
7  }
```

- 2.2 (4 Punkte) Stellen Sie das Programm aus 2.1 semantisch äquivalent mit 2 alternativen C-Schleifen dar.
- 2.3 (10 Punkte) Definieren Sie in der Programmiersprache C eine Struktur für Personen. Eine Person besteht aus den folgenden Elementen: Vorname und Nachname zusammen als eine Zeichenkette (max. 64 Zeichen), ledig (ja/nein), Alter (in Jahren), Zeiger auf einen Erziehungsberechtigten (ebenfalls vom Typ Person) und Kontostand (in €, ganzzahlig, kann negativ sein). Geben Sie den Elementen sinnvolle Variablennamen sowie adäquate Datentypen (speicherschonend). Erstellen Sie eine Variable, die maximal 5 Personen enthalten kann und eine weitere einzelne Variable vom Typ Person. Initialisieren Sie letztere mit beliebigen Werten. Angenommen eine Person ist sehr reich (Konto > ca. 5 Mrd.€). Erläutern Sie kurz, ob und warum dies zu Problemen führen könnte und wie man dieses Problem ggf. lösen kann.

Aufgabe 3 - C und SQL Programmierung

(16 Punkt(e))

- 3.1** (8 Punkte) Das folgende C-Programm besitzt zahlreiche syntaktische und semantische Fehler. Ein paar dieser Fehler wurden vom Compiler erkannt und unter dem Quellcode ausgegeben. Erläutern Sie zunächst die Struktur der Compilermeldungen allgemein. Interpretieren Sie die Compilerausgaben und korrigieren Sie die Fehler. Finden Sie darüber hinaus auch die semantischen Fehler und geben deren Lösung an.

```
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 typedef struct {
5     int aValue;
6     char aCharacter;
7 } t_myStruct;
8
9 int toLower(t_myStruct *theStruct){
10     if(theStruct->aCharacter >= 65 && theStruct.aCharacter <= 90){
11         theStruct->aCharacter += 32;
12         return 1;
13     }else{
14         return 0;
15     }
16 }
17
18 void copyCharToInt(char c, int* newInteger){
19     *newInteger = (int)c;
20     return EXIT_SUCCESS;
21 }
22
23 int main(void) {
24     int i = 0;
25     t_myStruct oneStruct;
26     while(i>5){
27         printf("grossen Buchstaben eingeben:");
28         if(toLower(&oneStruct, 23)){
29             printf("Der Buchstabe ist jetzt %c\n", oneStruct.aCharacter);
30         }else{
31             printf("Der Buchstabe ist immernoch %.2f\n", oneStruct.
32                 aCharacter);
33         }
34         scanf("%c",&oneStruct.aCharacter);
35     }
36     copyCharToInt(oneStruct.aCharacter, oneStruct.aValue);
37     printf("Der ASCII-Wert des Buchstaben %2$c ist %1$i\n", oneStruct.
38         aValue, oneStruct.aCharacter);
39     return EXIT_SUCCESS;
40 }
```

```
1 main.c:36:38: warning: incompatible integer to pointer conversion
   passing 'int' to parameter of type 'int *'; take the address with
   & [-Wint-conversion]
2 main.c:28:26: error: too many arguments to function call, expected
   single argument 'theStruct', have 2 arguments
3 main.c:20:2: error: void function 'copyCharToInt' should not return
   a value [-Wreturn-type]
4 main.c:31:47: warning: format specifies type 'double' but the
   argument has type 'char' [-Wformat]
5 main.c:10:45: error: member reference type 't_myStruct *' is a
   pointer; maybe you meant to use '->'?
```

3.2 (8 Punkte) SQL - Gegeben sind die Relationen in den unten angefügten Tabellen. Erstellen Sie für die Aufgaben a-f je ein SQL-Statement.

- a) Ausgabe des Namens und des Gehaltes aller leitenden Mitarbeiter (keine mNr in Attribut "leiter").
- b) Löschen des Mitarbeiters "Schulze, A." (mNr ist bekannt).
- c) Ausgabe des Mitarbeiter- und Abteilungsname aller Mitarbeiter in Dresden (abtNr von Dresden nicht bekannt).
- d) Hinzufügen eines leitenden Mitarbeiters (Name: Jürgen, K. - Gehalt: 2100) in der Forschungsabteilung in Berlin (abtNr ist bekannt).
- e) Dem Mitarbeiter Richter, J. eine Gehaltserhöhung auf 3800 geben (mNr ist bekannt).
- f) Ausgabe der Namen aller Mitarbeiter mit Gehalt über 1900 in Dresden (abtNr unbekannt), absteigend sortiert nach Gehalt.

| mNr | name | leiter | gehalt | abteilung |
|-----|-------------|--------|--------|-----------|
| 1 | Müller, N. | | 2000 | 1 |
| 2 | Meier, H. | 1 | 2300 | 1 |
| 3 | Schulze, A. | 1 | 1500 | 1 |
| 4 | Kunze, T. | | 4000 | 2 |
| 5 | Richter, J. | | 3500 | 3 |
| 6 | Möller, A. | 4 | 2100 | 2 |

Tabelle: Mitarbeiter

| abtNr | name | ort |
|-------|-----------|---------|
| 1 | Vertrieb | Dresden |
| 2 | Forschung | Berlin |
| 3 | Lager | Leipzig |
| 4 | Forschung | Dresden |

Tabelle: Abteilung

Aufgabe 4 - Minimalmaschine

(15 Punkt(e))

Übersetzen Sie das folgende Programmfragment in Befehle der in der Vorlesung behandelten Minimalmaschine MiMa. Sie können alle in der Tabelle angegebenen Befehle verwenden. Beachten Sie, dass n jeden Integer-Wert annehmen kann. Vorher ist eine semantisch äquivalente Umstellung des C-Codes möglich. Angenommen nach Zeile 05 des Codes würde man eine Ausgabe der Variable a implementieren. Wie würde diese für $n = 5$ aussehen?

```
1  int n;  
2  int a = 1, b = 0;  
3  while ( n ){  
4      if ( 0 < n ) {  
5          a = b + a;  
6          b++;  
7      }  
8      n--;  
9  }
```

| Instr. | Bedeutung | Instr. | Bedeutung |
|--------|------------------|--------|---|
| ADD n | acc=acc + mem[n] | JMP a | springe zu a |
| AND n | acc=acc & mem[n] | JMN a | springe zu a, wenn acc < 0 |
| OR n | acc=acc mem[n] | EQL n | wenn acc==mem[n],dann acc:=-1; sonst acc:=0 |
| XOR n | acc=acc ^ mem[n] | HALT | Prozessor anhalten |
| LDV n | acc=mem[n] | NOT | acc= acc (bitweise Komplement) |
| STV n | mem[n]=acc | RAR | acc um 1 Bit nach rechts rotieren |
| LDC n | acc=n | SUB n | acc=acc - mem[n] |
| MUL n | acc=acc * mem[n] | DIV n | acc=acc / mem[n] |

Aufgabe 5 - Ja/Nein Fragebogen

(20 Punkt(e))

Geben Sie bitte an, ob die folgenden Aussagen richtig oder falsch sind. Für jede richtig beantwortete Aussage gibt es einen Punkt und für eine falsch beantwortete Aussage wird ein Punkt abgezogen. Ein negatives Gesamtergebnis dieser Aufgabe wird nicht übertragen, d.h. in diesem Fall wird die Aufgabe mit 0 Punkten bewertet. Bitte schreiben Sie die Antworten direkt hier auf den Aufgabenzettel.

| # | Antwort | Frage |
|----|---------|--|
| 1 | | Ein Hyperlink ist eine bidirektionale hochgeschwindigkeits Punkt-zu-Punkt Verbindung zwischen Computerprozessoren. |
| 2 | | <code>sizeof(char) == 1</code> |
| 3 | | Jede primitiv rekursive Funktion lässt sich auch iterativ darstellen. |
| 4 | | Ein Linker wandelt Quellcode in maschinenlesbare Befehle um. |
| 5 | | <code>A && !B && (C && B) && !A</code> |
| 6 | | <code>0xBEEF + 01010101 == 1011 1111 0100 0100</code> |
| 7 | | Eine Flipflop-Schaltung kann aus zwei NAND-Gattern konstruiert werden. |
| 8 | | Die Minimalmaschine MiMa besitzt genau eine ALU. |
| 9 | | Mit Hilfe von <code>self</code> erfolgt in Swift der Zugriff auf die Basisklasse. |
| 10 | | Die Abkürzung des Farbraumes HSV steht für Human Saturated Value. |
| 11 | | Der Compiler findet alle semantischen Fehler im Programmquellcode. |
| 12 | | <code>((123456740 mod 1000) mod 100) mod 14 == 12</code> |
| 13 | | In der Programmiersprache Swift können nur 'optional'-Datentypen den Wert <code>nil</code> enthalten. |
| 14 | | <code>A ⊕ !A == 0</code> |
| 15 | | Allgemeine Turingmaschinen akzeptieren Typ-0 Sprachen. |
| 16 | | Der Begriff XML steht für Exclusive Markup Language. |
| 17 | | <code>a ∪ b</code> entspricht dem logischen Produkt von <code>a</code> und <code>b</code> . |
| 18 | | Der Informationsgehalt eines Symbols ist unabhängig von dessen Auftrittswahrscheinlichkeit. |
| 19 | | Quicksort ist stets effektiver als Bubblesort. |
| 20 | | Die Wurzel aus 2 ist genau 1,414. |

Aufgabe 6 - Swift

(20 Punkt(e))

6.1 (14 Punkte) Implementieren Sie eine Klasse mit dem Namen *Point*, die zwei Koordinaten (x, y) enthält. Die *Point*-Klasse soll des Weiteren eine Methode *getDistanceTo()* besitzen, die den Abstand zu einem anderen Punkt berechnet und zurückgibt. Als Parameter erwartet diese Methode einen Punkt. Implementieren Sie eine Klasse mit dem Namen *Circle*, welche den Mittelpunkt (vom erstellten Typ *Point*) des Kreises und seinen Radius enthält. Außerdem soll eine Methode *didCollide()* erstellt werden, die einen anderen Kreis als Parameter übergeben bekommt und eine Kollision feststellen kann. Zur Berechnung der Kollision u.a. soll die *getDistanceTo()* Methode der Klasse *Point* verwendet werden. Wird eine Kollision festgestellt, so soll *true* zurückgegeben werden, anderenfalls *false*.

HINWEIS: Die Quadratwurzel lässt sich in Swift mit der *sqrt()*-Funktion berechnen.

6.2 (6 Punkte) Schreiben Sie ein ausführbares Programm, das ein Array mit 10 Kreisen füllt. Die Kreise sollen zufällige Mittelpunktskoordinaten erhalten. Die *rand()* Funktion liefert dazu eine Pseudozufallszahl vom Typ *Int32* (`var zufallsZahl = rand()`).

Prüfen Sie dann alle Kreise paarweise auf Kollision.

Viel Erfolg!

Anzahl der abgegebenen Blätter:

| Aufgabe: | Punkte max: | Punkte erreicht: | Bemerkung: |
|-----------------|--------------------|-------------------------|-------------------|
| 0 | 1 | | |
| 1.1 | 1 | | |
| 1.2 | 4 | | |
| 1.3 | 2 | | |
| 1.4 | 5 | | |
| 2.1 | 2 | | |
| 2.2 | 4 | | |
| 2.3 | 10 | | |
| 3.1 | 8 | | |
| 3.2 | 8 | | |
| 4 | 15 | | |
| 5 | 20 | | |
| 6.1 | 14 | | |
| 6.2 | 6 | | |
| Gesamt: | 100 | | |
| | | Note: | |