

Montag, den 04.02.2008

Prof. Dr.-Ing. Andreas Meisel

## Klausur MN

Name

Matrikel-Nummer

### Hinweise:

- 1.) Tragen Sie in obige Felder Ihren Namen und Ihre Matrikelnummer ein.
- 2.) Zusätzliche Lösungsblätter versehen Sie bitte mit **Namen und Matrikelnummer**.  
Nehmen Sie zur Bearbeitung einer Aufgabe jeweils ein neues Blatt.
- 3.) Vermerken Sie in den vorgesehenen Lösungsfeldern der Aufgabenblätter, dass ein Zusatzblatt existiert.
- 4.) Zur Bearbeitung stehen **135 Minuten** zur Verfügung.
- 5.) **Erlaubte Hilfsmittel:**  
4 Blatt (DIN A4), einseitig (persönlich) handschriftlich beschrieben, alle Blätter müssen mit Name und Matrikelnummer beschriftet sein.  
Sonst keine weiteren Hilfsmittel (Taschenrechner, Notebooks, Handy's).

Übersicht zur Bewertung der Aufgaben.		
Aufgabe	Punkte	
01	15	
02	20	
03	15	
04	20	
Summe $\cong$	70	

**Aufgabe 1** (Grundlagenwissen)

[15 Punkte]

<p>Geben Sie als <u>Hexadezimalwert</u> an:</p> <p>10010100110111101<sub>B</sub></p>	
<p>Geben Sie den <u>Dezimalwert</u> der Hexadezimalzahl AA<sub>H</sub> an.</p>	
<p>Wie lautet der Dezimalwert der Binärzahl 0110 0101<sub>B</sub></p> <p>a) als vorzeichenlose Zahl</p> <p>b) als vorzeichenbehaftete Zahl (8-bit)</p>	
<p>Wie lautet die Binärdarstellung (8-Bit) der vorzeichenbehafteten Zahl -77<sub>D</sub> ?</p>	
<p>Gegeben ist eine <u>vorzeichenlose</u> Festkommazahl mit 6 Vorkomma- und 10 Nachkommastellen.</p> <p>Welche kleinste (ohne 0) und größte Zahl ist damit darstellbar (Darstellung als Bruch !).</p>	<p>kleinste Zahl:</p> <p>größte Zahl:</p>
<p>Berechnen Sie (8-bit-Zahl) :</p> <p>Wird das Carry-Flag gesetzt?</p> <p>Wird das Overflow-Flag gesetzt?</p>	<pre> 1 1 0 0 0 1 1 1 +0 1 1 0 1 1 0 0 ----- </pre>
<p>Berechnen Sie (8-bit-Zahl):</p> <p>Wird das Carry-Flag gesetzt?</p> <p>Wird das Overflow-Flag gesetzt?</p>	<pre> 1 1 1 1 0 0 1 0 -1 1 1 0 1 0 1 1 ----- </pre>
<p>Gegeben ist das C-Codefragment:</p> <pre> char a=50, Erg; Erg = a*6; </pre> <p>Welchen Dezimalwert hat Erg nach der Addition?</p>	
<p>Geben Sie die Assemblerbefehle an, um nach LAB zu springen, wenn in %al eine <u>gerade Zahl</u> steht.</p>	
<p>Geben Sie eine Assembler-Befehlssequenz an, mit der Sie nach Label LAB springen, wenn in %al die Bits 0..2 den Wert 1 haben und die Bits 5..7 den Wert 0.</p>	



**Aufgabe 3** (Unterprogramme, Parameterübergabe, Stack)

[15 Punkte]

Schreiben Sie ein strukturiertes Assembler-Unterprogramm, welches das folgende C-Unterprogramm ersetzt:

```
int Test(int x, int sh)
{
    char ret=0;

    if (x<0)          ret = -1;
    else if (x>=32)  ret = +1;
    else              ret = x + sh;

    return ret ;
}
```

1. Die Parameter `x` und `sh` sollen über den Stack übergeben werden.
2. Das Ergebnis soll über `%eax` zurückgegeben werden.
3. Alle verwendeten Register sollen gerettet und restauriert werden.

**Anm.:** Verwenden Sie das Lösungsfeld auf der nächsten Seite.

\*\*\*\*\* Unterprogramm TEST \*\*\*\*\*

Test:

**Aufgabe 4** (Fragen zur Programmiersprache C)

[20 Punkte]

<p>Schreiben Sie ein äquivalentes Programm, welches <u>kein</u> "break" oder "goto" enthält.</p> <pre>y=0; for(i=0; i&lt;500; i++) {     y=y+func(i);     if(y&gt;=12) break; }</pre>	
<p>Geben sie zu folgender Schleife eine äquivalente <u>for-Schleife</u> an:</p> <pre>int i = 0; do{     printf("i=%d", ++i); } while (i &lt; 10);</pre>	
<p>Welchen Wert hat a nach folgender Sequenz?</p> <pre>int x = 21, y = 9; double a; a = x / y;</pre>	
<p>Welchen Wert hat a nach folgender Sequenz?</p> <pre>char x=-34, a; if (x&gt;=0 &amp;&amp; x&lt;=10) a=1; else if (x&gt;10    x&lt;-34) a=2; else a=3;</pre>	
<p>Rücken Sie das folgende (ganz schlecht eingerückte) Programm so ein, dass die Strukturierung deutlich wird.</p> <pre>int z = 5, y = -1, x = 0;  if(x&gt;0) z=0; else if(x==0) z=1; if (y&lt;=0) z=2; else z=3; else z=4 ;</pre>	
<p>Was wird ausgedruckt?</p> <pre>#define SHL(A,B) A&lt;&lt;B int a=3, b=2, c=1; printf("%d",SHL(a,b+c));</pre> <p>Was könnte man besser machen?</p>	

<p>Ein Unterprogramm zur Berechnung des Skalarproduktes zweier 3-elementiger Vektoren <u>ist gegeben</u> und hat den folgenden Prototyp:</p> <pre>int SkalProd(int *v1, int *v2);</pre> <p>Gegeben sind weiter der Vektor A und die Matrix B:</p> <pre>int A[] = {12,3,5}; int B[][3] = { {22,0,6},                {11,2,5} };</pre> <p>Geben Sie den <u>Aufruf</u> zur Berechnung des Skalarproduktes des Vektors A und der zweiten Zeile von B an.</p>	<pre>/* Aufruf des Unterprogramms SkalProd */ Erg = // xxx // xxx // xxx // xxx // xxx</pre>
<p>Gegeben ist eine Matrix:</p> <pre>char D[][3] = { {12,3,5},                {22,0,6},                {11,2,4},                {1,3,55} };</pre> <pre>int *pD[] = {D[3],D[2],D[1],D[0]};</pre> <pre>int **ppD = pD;</pre> <p>Die Matrix D beginne bei Adresse 1000 (dezimal). Die Zeigermatrix pD beginne bei Adresse 2000 (Länge eines Zeigers: 4 Byte). ppD stehe bei Adresse 3000.</p> <p>Vervollständigen Sie nebenstehende Tabelle:</p>	<pre>D[1] = &amp;D[2][2] = pD[3] = pD = ppD[0][0] = ppD =</pre>
<p>Gegeben ist folgende Initialisierung:</p> <pre>char *Tab[] = {"ab","def","ghi"};</pre> <pre>char *Temp;</pre> <p>Geben Sie <u>eine Befehlssequenz</u> an, um den 2. und 3. String in Tab zu tauschen (d.h. Zeigertausch).</p>	
<p>Vervollständigen Sie die nebenstehende <u>Programmdefinition</u> des Unterprogramms LastChar(...).</p> <p>Der String soll „per reference“ an das Unterprogramm übergeben werden.</p> <p>Das Unterprogramm soll das letzte Zeichen des Strings (also das Zeichen vor dem Nullterminator) zurückgeben.</p>	<pre>char LastChar( { } }</pre>