Prof. Dr. K. Wüst
FH Gießen–Friedberg, FB MNI

WS 2001
Studiengang Informatik

Klausur zu Maschinennahe Programmierung

23.1.2002

Nachname:
Vorname:
Matrikelnummer:

Punkteverteilung

Aufgabe	Punkte	erreicht
1	8	
2	8	
3	10	
4	8	
5	6	
Summe	40	
Note	–	

Auf die Rückseiten schreiben oder Zusatzblätter abgeben, wenn der Platz nicht reicht!

Aufg.1) Geben Sie den Inhalt der Register AX, BX, CX, DX nach der Ausführung der untenstehenden Befehle in hexadezimaler Form an. Die Tabelle ist nur für Ihre Notizen gedacht und muss nicht ausgefüllt werden.

Befehl	AX	BX	CX	DX
mov ax,10				
imul ax,3				
dec ax				
shr ax,1				
xor bx,bx				
rcl bx,1				
mov cx,13				
and cx,7				
mov dx,0				
l1: add dx,cx				
loop l1				
Ergebnis				

Aufg.2) Schreiben Sie in Assembler ein Unterprogramm, das einen Speicherdump, d.h. eine Darstellung des Inhaltes eines bestimmten Hauptspeicherabschnittes, auf den Bildschirm ausgibt. Es wird eine 16-Bit-Umgebung vorausgesetzt und die Zeichen sollen mit Funktion 02h von INT 21h ohne weitere Umsetzung ausgegeben werden. Vor Aufruf von Funktion 02h muss das auszugebende Zeichen in DL hinterlegt werden. Die Schnittstelle soll sein:

```
Vor Aufruf:
BX: Zeiger auf erstes Zeichen des Hauptspeicherabschnittes
CX: Anzahl der auszugebenden Zeichen
```

Aufg.3) Schreiben Sie ein Unterprogramm, das für zwei 32-Bit-Muster feststellt auf wie vielen Positionen diese Bitmuster übereinstimmen. Ein Beispiel (für 4-Bit-Muster): Die Bitmuster 0110b und 1010b stimmen auf zwei Positionen überein (den beiden letzten). Entwerfen und beschreiben Sie auch die Schnittstelle Ihres Unterprogramms!

Aufg.4) In einem Dreieck wird die Länge a aus der Länge b und dem Winkel x berechnet nach der Formel $a = b \sin x$. Alle Variablen sind 64-Bit-Gleitkommazahlen. Schreiben Sie ein Assembler-Programm, das a nach der obigen Formel aus b und x berechnet!

Aufg.5) Schreiben Sie in Inline-Assembler den Rumpf einer C-Funktion, die die Inhalte zweier Integer-Variablen vertauscht.

```
void Vertausche(int *zahl1, int *zahl2) {



}
```