

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

Hiermit bestätige ich, dass ich die Übungsleistungen als Voraussetzung für diese Klausur in folgender Übung erfüllt habe.

Jahr: _____ **Übungsleiter:** _____ **Unterschrift:** _____

1. Aufgabe (/ 12 Pkt.)

Was liefert das folgende Programm an Bildschirmausgaben?

```
#include <iostream>
using namespace std;

int main()
{
    const int size = 25;
    int a = 1, b = 10, z = 1;
    while ( a + b < size )
    {
        cout << z << ". Zeile: " << (b - a) * 2 << endl;
        a = a + 2;
        b++;
        z++;
    }
}
```

(4 Punkte)

1. Zeile: 18
2. Zeile: 16
3. Zeile: 14
4. Zeile: 12
5. Zeile: 10

```
z = 1;
for ( a = 3; a < 8; a++ )
{
    cout << "Ausgabe " << z << ": ";
    for ( b = size; b > 0; b = b - 5 )
    {
        cout << (a + b) * 2 << " ";
    }
    cout << endl;
    z++;
}
cout << endl;
```

(8 Punkte)

Ausgabe 1:	56	46	36	26	16
Ausgabe 2:	58	48	38	28	18
Ausgabe 3:	60	50	40	30	20
Ausgabe 4:	62	52	42	32	22
Ausgabe 5:	64	54	44	34	24

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

2. Aufgabe (/ 25 Pkt.)

- a) Formulieren Sie eine Funktionsdeklaration, die für ein gegebenes Array **V** mit **n** reellen Komponenten **V_i** den Mittelwert **mw** und die Standardabweichung **stda** ermittelt. Ein- und Ausgabegrößen sollen als Parameter (nicht als Return-Wert) übergeben werden. (4 Punkte)

```
void funktion( double V[], int n, double *mw, double *stda );
```

- b) Man gebe die Implementierung dieser Funktion an. Zur Berechnung des Mittelwertes und der Standardabweichung verwende man die bekannten Formeln: (13 Punkte)

$$mw = \left(\sum_{i=1}^n v_i \right) / n$$

$$stda = \left(\sum_{i=1}^n (v_i - mw) \right) / n$$

```
void funktion( double V[], int n, double *mw, double *stda )
{
    int i;
    double summe = 0.0;
} } 3 Punkte

    for (i = 0; i < n; i++ )
    {
        summe = summe + V[i];
    }
    *mw = summe / n;
} } 5 Punkte

    summe = 0.0;
    for (i = 0; i < n; i++ )
    {
        summe = summe + (V[i] - *mw);
    }
    *stda = summe / n;
} } 5 Punkte
}
```

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

- c) Implementieren Sie das main-Programm mit einem Funktionsaufruf Ihrer Funktion, so dass mit deren Hilfe für die **50** gegebenen Werte einer aktuellen Messreihe die Größen **mw** und **stda** bestimmt werden. Die Messreihe sei in einem Datenarray **v [100]** gespeichert. Die Eingabe der Messwerte ist nicht gefordert!
(8 Punkte)

```
#include "Funktionen.h"  
#include <iostream>  
using namespace std;
```

} **2 Punkte**

```
int main()  
{  
    const int maxAnzahl = 100;  
    int n = 50;  
    double mw = 0, stda = 0;  
    double V[maxAnzahl] = { /* 50 Messwerte */ };
```

} **2 Punkte**

```
// Aufruf der Funktion  
funktion( V, n, &mw, &stda );
```

} **2 Punkte**

```
cout << "Der Mittelwert ist          = " << mw << endl;  
cout << "Die Standardabweichung ist = " << stda << endl;  
cout << endl;
```

} **2 Punkte**

```
return 0;  
}
```

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

3. Aufgabe (/ 10 Pkt.)

- a) Geben Sie die binäre Darstellung des dezimalen Zahlenwertes 43 an (inklusive Rechenweg).
Wie wird diese Zahl bei einem Integer-Format mit 2 Byte im Rechner gespeichert? (3 Punkte)

$$\begin{array}{r} 43 / 2 = 21 \text{ R } 1 \\ 21 / 2 = 10 \text{ R } 1 \\ 10 / 2 = 5 \text{ R } 0 \\ 5 / 2 = 2 \text{ R } 1 \\ 2 / 2 = 1 \text{ R } 0 \\ 1 / 2 = 0 \text{ R } 1 \end{array}$$

$$\Rightarrow 101011 \quad \Rightarrow 00000000 \ 00101011$$

- b) Geben Sie die Darstellung des dezimalen Zahlenwertes 82 im Zahlensystem mit der Basis 6 an
(inklusive Rechenweg und Proberechnung) (3 Punkte).

$$\begin{array}{r} 82 / 6 = 13 \text{ R } 4 \\ 13 / 6 = 2 \text{ R } 1 \\ 2 / 6 = 0 \text{ R } 2 \end{array}$$

$$\Rightarrow 214_6$$

$$\Rightarrow \text{Probe: } 4 * 6^0 + 1 * 6^1 + 2 * 6^2 = 4 + 6 + 72 = 82$$

- c) Berechnen Sie binär (inklusive Rechenweg) und schreiben Sie das Ergebnis als Binär- und als Dezimalzahl. Führen Sie anschließend eine Proberechnung durch, um das Ergebnis zu bestätigen.

$$101011 * 111$$

(4 Punkte)

$$\begin{array}{r} 101011 * 111 \\ \hline 101011 \\ 101011 \\ 101011 \\ \hline 100101101 \end{array}$$

$$\text{Probe: } 1 + 4 + 8 + 32 + 256 = 301$$

$$101011_2 = 43_{10} \quad \text{und} \quad 111_2 = 7_{10} \\ 43 * 7 = 301$$

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

4. Aufgabe (/ 4 Pkt.)

Gegeben ist folgende Wahrheitstabelle:

a	b	c	f1(a,b,c)	f2(a,b,c)	f3(a,b,c)	f4(a,b,c)
0	0	0	1	1	0	0
0	0	1	0	1	1	1
0	1	0	0	0	0	1
0	1	1	1	1	1	0
1	0	0	0	1	0	0
1	0	1	0	1	1	1
1	1	0	1	0	0	1
1	1	1	0	1	1	0

Geben Sie bitte die Boole'schen Funktionen an, die die Bedingungen erfüllen, wobei nur die booleschen Operatoren "und", "oder", "nicht" erlaubt sind.

f1 (a,b,c) = **((nicht a) und (nicht b) und (nicht c)) oder ((nicht a) und b und c) oder (a und b und (nicht c))**

f2 (a,b,c) = **(nicht b) oder c**

f3 (a,b,c) = **c**

je 1 Punkt

f4 (a,b,c) = **((nicht b) und c) oder ((b und (nicht c))**

5. Aufgabe (/ 6 Pkt.)

Skizzieren und beschreiben Sie kurz die Wirkungsweise der System-Software-Komponenten (Compiler, Debugger, Editor, Linker), die man benötigt, um von einem Quellprogramm-Entwurf zum lauffähigen Maschinenprogramm zu kommen. Nennen Sie jeweils Beispiele und beachten Sie dabei die richtige Reihenfolge.

Editor: Erstellen/Modifizieren des Programmtextes (Header- und Quellcode-Dateien).

Beispiele: Editor in Visual C++ Express, notepad

je 1,5 Punkte

Compiler: Programmtext (Quellcodedatei) übersetzen, dabei Objectcode erzeugen. Nur ein vollständig fehlerfreies Programm kann in Objectcode übersetzt werden.

Beispiele: Compiler in Visual C++ Express, gcc, g++, cc

Linker: Ausführbares Programm erzeugen (aus den zuvor erzeugten Objectcode-Dateien)

Beispiele: Linker in Visual C++ Express, gcc, g++, cc

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

Debugger: Programm ausführen und testen. Der Debugger ist nicht nur zur Lokalisierung von Programmierfehlern, sondern auch zur Analyse eines Programms durch Nachvollzug des Programmablaufs hilfreich
Beispiele: Debugger in Visual C++ Express, gdb, dbx

6. Aufgabe (/ 20 Pkt.)Gegeben ist das folgende Unterprogramm **funktion()**:

```
void funktion( int & zahl )
{
    int temp = 2;
    bool ausgabe = false;

    while ( ( temp <= zahl ) && ( false == ausgabe ) )
    {
        if ( 0 == ( zahl % temp ) )
        {
            zahl = zahl / temp;
            funktion( zahl );
            cout << temp << " * ";
            ausgabe = true;
        }
        else
        {
            temp++;
        }
    }
}
```

- a) Was wird ausgegeben, wenn das Programm mit einen Wert **24** für die Variable **zahl** ausgeführt wird? (8 Punkte)

3 * 2 * 2 * 2 *

- b) Wie nennt man das Ergebnis bzw. welche mathematische Funktion ist hier programmiert? (2 Punkte)

absteigende Primfaktorenzerlegung

- c) Welche besondere Programmiertechnik wurde im Unterprogramm **funktion()** verwendet? (1 Punkt)

Rekursion

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

- d) Schreiben Sie eine vollständige Main-Funktion (Hauptprogramm), die eine int-Eingabe bewirkt und die oben stehende Funktion mit der Zahl aufruft. (9 Punkte)

```
#include "Funktionen.h"  
#include <iostream>  
using namespace std;
```

} **3 Punkte**

```
int main()  
{  
    int zahl = 0;  
  
    cout << "Bitte ganzzahligen Wert eingeben: ";  
    cin >> zahl;
```

} **3 Punkte**

```
// Aufruf der Funktion  
funktion( zahl );
```

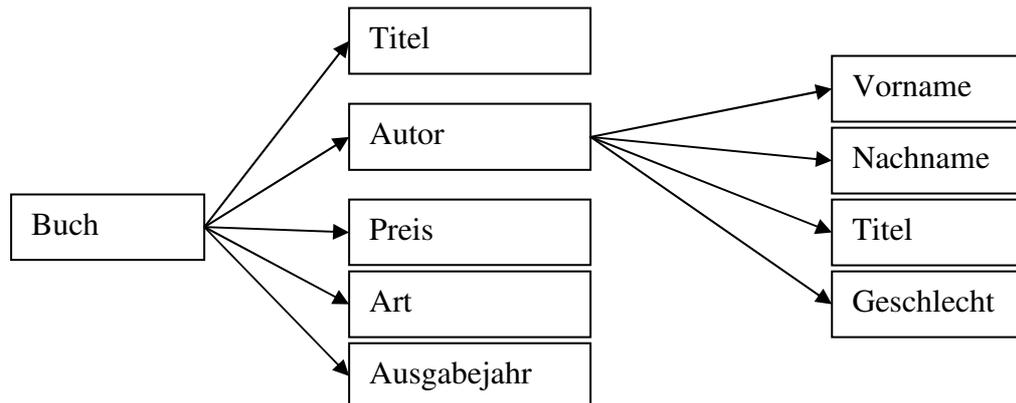
} **3 Punkte**

```
return 0;  
}
```

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

7. Aufgabe (/ 23 Pkt.)

Für eine Buchhandlung sollen für maximal 500 Bücher Informationen gespeichert werden, die alle nach der gleichen Weise wie folgt strukturiert sind:



Dabei gelten folgende Beschreibungen:

Titel ist max. 50 Zeichen lang

Autor enthält die vier Elemente:

Vorname ist max. 35 Zeichen lang

Nachname ist max. 35 Zeichen lang

Titel Aufzählungstyp mit Wertevorrat: Kein, Dr., Prof., Dipl.Ing.,

Geschlecht Aufzählungstyp mit Wertevorrat: maennlich, weiblich

Preis ist eine reelle Zahl

Art Aufzählungstyp mit Wertevorrat: Taschenbuch, Hardcover, Softcover

Ausgabejahr ist eine ganze Zahl

a) Beschreiben Sie in C/C++ diesen Datentyp vollständig (alle notwendigen Angaben) (12 Punkte)

```
enum myTitle      { Kein, Dr, Prof, DiplIng };
enum myGeschlecht { maennlich, weiblich };
enum myArt        { Taschenbuch, Hardcover, Softcover };
```

3 Punkte

```
struct myAutor {
    char    Vorname[35];    // oder string Vorname;
    char    Nachname[35];  // oder string Nachname;
    myTitle Titel;
    myGeschlecht Geschlecht;
};
```

4 Punkte

```
struct Buch {
    char    Titel[50];    // oder string Titel;
    myAutor Autor;
    double  Preis;
    myArt   Art;
    int     Ausgabejahr;
};
```

5 Punkte

Nachname: _____ Vorname: _____ Matr.-Nr.: _____ Punkte: _____

- b) Zeigen Sie in einem Hauptprogramm, wie Ihr Datentyp instantiiert wird und zeigen Sie an untenstehendem Beispiel, wie ein neues Buch an der 27 Stelle der Liste in die Variablen eingetragen wird: (11 Punkte)

Titel (des Buches): Tintenherz
Vorname: Horst
Nachname: Schimanski
Titel (des Autors): Dr.
Geschlecht: maennlich
Preis: 27,45
Art: Hardcover
Ausgabejahr: 2005

```
Buch buchInfo[500];

strcpy( buchInfo[26].Titel, "Tintenherz" );
// oder falls string Titel;
// buchInfo[26].Titel = "Tintenherz";
strcpy( buchInfo[26].Autor.Vorname, "Horst" );
// oder falls string Vorname ;
// buchInfo[26].Autor.Vorname = "Horst";
strcpy( buchInfo[26].Autor.Nachname, "Schimanski" );
// oder falls string Nachname ;
// buchInfo[26].Autor.Nachname = "Schimanski";
buchInfo[26].Autor.Titel = Dr;
buchInfo[26].Autor.Geschlecht = maennlich;
buchInfo[26].Preis = 27.45;
buchInfo[26].Art = Hardcover;
buchInfo[26].Ausgabejahr = 2005;
```