



**Klausur zur Vorlesung
Modellierung und Programmierung
Wintersemester 2010/2011
Termin 25.03.2011**

Beachten Sie die folgenden Hinweise:

- **Anzahl der Fragen: 25 auf 7 Blättern, Dauer der Klausur: 100 Minuten.**
- **Die Klausur ist bestanden bei höchstens 10 Fehlern.** Zur Benotung siehe Tabelle.
- Bei jeder Frage ist **genau eine Antwortalternative richtig.**
Jede Bearbeitung, die im Ankreuzen einer falschen Antwort besteht, wird als Fehler gewertet.
- Halten Sie bitte Ihren Studentenausweis bereit.
- Verwenden Sie ausschließlich das ausgeteilte Papier.
- **Tragen Sie auf jedes Blatt Ihren Namen und Ihre Immatrikulationsnummer an den dafür vorgesehenen Stellen ein.**
- **Verwenden Sie zum Schreiben weder (Druck-)Bleistift noch Rotstift.** Bei Nichtbeachtung wird die Klausur nicht gewertet.
- Es sind **keine Hilfsmittel zugelassen.** Nichtbeachtung führt zum Ausschluss von der Klausur.
- **Bekanntgabe der Ergebnisse:** Dienstag, 29.03.2011 durch Aushang in Geb. E2.4, im 3. Stock von Geb. E1.1 und auf der Website der Veranstaltung.
- **Klausureinsicht:** Mittwoch, 30.03.2011, 11.00-12.00 Uhr in Geb. E1.1, Raum 323.1.

Viel Glück!

Teilnehmer/Teilnehmerin

Name:

Vorname:

Matr.-Nr.:

Notengrenzen:

| Note | Fehler |
|--------------------|----------|
| sehr gut (1.0) | ≤ 1 |
| sehr gut (1.3) | 2 |
| gut (1.7) | 3 |
| gut (2.0) | 4 |
| gut (2.3) | 5 |
| befriedigend (2.7) | 6 |
| befriedigend (3.0) | 7 |
| befriedigend (3.3) | 8 |
| ausreichend (3.7) | 9 |
| ausreichend (4.0) | 10 |

Bei 11 oder mehr Fehlern ist die Klausur nicht bestanden.

Aufgabe 1 (Zahldarstellung im Binärformat).

Wenn die Dezimaldarstellung einer Gleitpunktzahl durch 99.5 gegeben ist, so lautet ihre Binär-
darstellung:

- (a) 101.01
- (b) 1111.1
- (c) 101010.1
- (d) 110011.1
- (e) 1100011.1

Aufgabe 2 (Gleitpunktzahlen).

Betrachten Sie normalisierte binäre Gleitpunktzahlen mit der Darstellung (s, f, E) , wobei $s \in \{0, 1\}$ ein Vorzeichenbit, f eine Mantisse mit Länge 23 Bits und $E \in \{-126, \dots, 127\}$ den Exponenten bezeichnet. Welche Aussage ist falsch?

- (a) Mit dieser Darstellung ist nur eine endliche Menge von Zahlen darstellbar.
- (b) Alle Zahlen mit Betrag größer als 2^{128} sind nicht darstellbar.
- (c) Alle Zahlen mit Betrag kleiner als 2^{-127} sind nicht darstellbar.
- (d) $\frac{1}{3} = \sum_{k=1}^{\infty} 2^{-2k}$, die Zahl $\frac{1}{3}$ ist nicht darstellbar als (s, f, E) .
- (e) $1 = \sum_{k=1}^{\infty} 2^{-k}$, aber die Zahl 1 ist darstellbar als (s, f, E) .

Aufgabe 3 (Rechnerarithmetik).

Es seien \mathbb{G} die Menge aller Gleitpunktzahlen aus Aufgabe 2, \oplus , \odot und \oslash Gleitpunktoperationen die der Addition, Multiplikation und Division in \mathbb{G} entsprechen. Welche Aussage ist falsch?

- (a) Besitzt das Ergebnis einer Operation einen Exponenten größer als E_{max} , so spricht man von einem Gleitpunktunterlauf.
- (b) Es existieren $\beta \in \mathbb{G}$, $\beta > 0$, sodass $1.0 \oplus \beta = 1.0$.
- (c) Für alle $a \in \mathbb{G}$, $0 \leq a \leq 2^{E_{max}}$ gibt es ein $b \in \mathbb{G}$, sodass $a \oplus b = 0$.
- (d) Es existieren $\gamma \in \mathbb{G}$ mit $1 < \gamma < 10$, sodass $1.0/\gamma \neq 1.0 \oslash \gamma$.
- (e) Die Anzahl der Elemente in \mathbb{G} ist endlich.

Aufgabe 4 (Deklaration von Variablen).

Welche der folgenden Variablendeklarationen ist syntaktisch falsch?

- (a) `signed char lway;`
 - (b) `unsigned short int b;`
 - (c) `const char c_c;`
 - (d) `static int b1c;`
 - (e) `double *f[32];`
-

Aufgabe 5 (Typumwandlungen).

Welche Ausgabe liefert das folgende Programm?

```
#include <stdio.h>
int main(void) {
    int i,j;
    double x;
    i=10 + 1/10;
    j=49/10;
    x=(double) (i/j);
    printf("%lf ",x);
    x=((double) i)/j;
    printf("%lf\n",x);
    return 0;
}
```

- (a) 2.200000 2.200000
 - (b) 2.200000 2.000000
 - (c) 3.000000 2.500000
 - (d) 2.000000 2.500000
 - (e) 1.000000 1.000000
-

Aufgabe 6 (Logische Ausdrücke).

Welche Ausgabe liefert das folgende Programm?

```
#include <stdio.h>
int main(void) {
    int i,j;
    i= !((2010 || 2011) || 2012);
    j= !i && (1 || 0);
    printf("%d %d\n",i,j);
    return 0;
}
```

- (a) -2012 2013
- (b) 2012 2013
- (c) 1 0
- (d) 0 1
- (e) 0 0
-

Aufgabe 7 (Logische Ausdrücke).

Welche Ausgabe liefert das folgende Programm?

```
#include <stdio.h>
int main(void) {
    int i=0,j=0,k=0;
    if(0) i=(++j==k++)&&(k--j)+10*(k--);
    if(i) i=(j==k);
    j= !( 0 && (k=3));
    printf("i=%d j=%d k=%d\n",i,j,k);
    return 0;
}
```

- (a) i=0 j=0 k=0
- (b) i=1 j=0 k=3
- (c) i=0 j=1 k=0
- (d) i=0 j=1 k=3
- (e) i=1 j=1 k=0
-

Aufgabe 8 (Zeiger und Felder).

Betrachten Sie das zweidimensionale Feld

```
int G[3][4]={{1,2,4,8}, {7,5,3,1}, {0,6,12,18}};
```

Welche der folgenden Ausdrücke ist nicht Null?

- (a) $G[2][2]-12$
- (b) $*(G + 1) - G[0][1]$
- (c) $*(G+2)[0]$
- (d) $**G$
- (e) $*(G[1]) - 7$

Aufgabe 9 (Gültigkeitsbereiche von Variablennamen).

Welche Ausgabe erzeugt das folgende Programm?

```
#include <stdio.h>
int a=10;
void f1( void) { printf("%d ",a); a++; }
void f2(int a) { printf("%d ",a); a++; }
int main(void) {
    f1();
    f2(a);
    printf("%d\n",a);
    return 0;
}
```

- (a) 10 11 11
 - (b) 10 11 12
 - (c) 10 12 12
 - (d) 11 11 11
 - (e) 12 10 11
-

Aufgabe 10 (Was tut die Schleife?).

Es seien i , N und $wert$ vom Typ `int`, wobei in N ein positiver Wert gespeichert sei. Welchen Wert besitzt die Variable $wert$ nach Durchlaufen der folgenden Schleife?

```
wert = 0;
for(i= -3*N; i<=5*N; i+=N) {
    wert += i;
}
```

- (a) N
 - (b) $4*N$
 - (c) $5*N$
 - (d) $8*N$
 - (e) $9*N$
-

Aufgabe 11 (Was tut die Schleife?).

Welche Ausgabe erzeugt das folgende Programm?

```
#include <stdio.h>
int main(void) {
    int i=2011;
    for(i=3;i<=8;i*=2) i--;
    printf("%d\n",i);
    return 0;
}
```

- (a) 7
 - (b) 8
 - (c) 9
 - (d) 10
 - (e) 2011
-

Aufgabe 12 (printf-funktion).

Welche Ausgabe erzeugt das folgende Programm?

```
#include <stdio.h>
int main(void) {
    float a=11.0;
    printf(stdout, "a=%f\n", a);
    return 0;
}
```

- (a) a=11
 - (b) a=11.0
 - (c) a=11.000000
 - (d) a=+11.000000000
 - (e) Das Programm ist syntaktisch nicht korrekt und kann gar nicht erst übersetzt werden.
-

Aufgabe 13 (Umgang mit Dateien).

Die Textdatei namens `daten.txt` soll über den Dateideskriptor `f` zum lesen geöffnet werden. Welche der folgenden Anweisungen ist hierzu geeignet?

- (a) `daten = fopen(f, "w");`
 - (b) `f = fopen(daten.txt, r);`
 - (c) `f = fopen("daten.txt", r);`
 - (d) `f = fopen("daten.txt", "r");`
 - (e) Keine der Anweisungen (a) – (d) leistet das Gewünschte.
-

Aufgabe 14 (Unveränderbare Zeiger).

Die Deklaration von y lautet

```
int const * const y;
```

Welche der folgenden Operationen ist erlaubt?

- (a) `y++`
 - (b) `(*y)++`
 - (c) `(*y)+1`
 - (d) `y=y+4`
 - (e) `(*y)=(*y)+4`
-

Aufgabe 15 (Zeigerarithmetik).

Welche Ausgabe erzeugt das folgende Programm?

```
#include <stdio.h>
int main(void) {
    int d[10], *p, i=0;
    for(p=d; p-d<10; p++)
        (*p)=p-d+1;
    p--;
    printf("%d\n",*p);
    return 0;
}
```

- (a) 0
 - (b) 1
 - (c) 9
 - (d) 10
 - (e) 100
-

Aufgabe 16 (Standardausgabeströme).

Welche der folgenden Aussagen stimmt nicht?

- (a) Über `stderr` erfolgen die Ausgaben gepuffert, aber die Ausgaben über `stdout` sind ungepuffert.
- (b) Die Funktion `printf` schreibt auf `stdout`.
- (c) Die funktion `scanf` liest aus `stdin`.
- (d) Die Ströme `stdout` und `stderr` sind Ausgabeströme.
- (e) Die Ströme `stdin`, `stdout` und `stderr` sind in Datei `stdio.h` deklariert.

Aufgabe 17 (Strukturen).

Welche Ausgabe erzeugt das folgende Programm?

```
#include <stdio.h>
struct Data {
    int k; char c[10];
} data1={1,"a"}, data2={2,"bbb"};
int main(void) {
    data2=data1;
    data1.k++;
    printf("%d %ld %s\n", data1.k, data2.k, data2.c);
    return 0;
}
```

- (a) 2 1 a
 - (b) 2 1 bbb
 - (c) 1 2 bbb
 - (d) 1 2 b
 - (e) 1 1 a
-

Aufgabe 18 (Kommandozeilenparameter).

Die Funktion `main` des Programms `prog` hat die folgende Deklaration

```
int main(int argc, char **argv)
```

Welche der folgenden Aussagen stimmt nicht?

- (a) `argc` speichert die Anzahl der beim Programmaufruf übergebenen Argumente, einschließlich des Programmnamens.
- (b) Wenn der Programmaufruf
`prog a b c d`
lautet, hat `argc` der Wert 4.
- (c) `argv` ist ein Feld von Strings, dessen Einträge die übergebenen Argumente sind.
- (d) Bei jedem Programmaufruf besitzt `argv[argc]` der Wert `NULL`.
- (e) Bei jedem Programmaufruf besitzt `argv[0][0]` der Wert `'p'`.

Aufgabe 19 (Strukturen).

```
#include <stdio.h>
typedef struct { double x,y; } Punkt;
void f1(Punkt* p) { p->x += p->y; }
void f2(Punkt p) { f1(&p); }
int main(void) {
    Punkt p={1.0, 1.0};
    f2(p);
    f1(&p);
    printf("%lf %lf\n", p.x, p.y);
    return 0;
}
```

- (a) 1.000000 1.000000
 - (b) 1.000000 2.000000
 - (c) 2.000000 1.000000
 - (d) 2.000000 2.000000
 - (e) 3.000000 1.000000
-

Aufgabe 20 (Komplexität).

Sei $A = \{A_{ij}\}_{i,j=1}^n \in \mathbb{R}^{n \times n}$ eine Matrix, wobei $A_{ij} = 0$ wenn $i + j = n + 1$ oder $i = 1$ oder $j = 1$.
Wieviele Speicherplätze benötigt man zur Speicherung von A ?

- (a) n
- (b) $2n$
- (c) $\frac{n(n+1)}{2}$
- (d) $n^2 - 3n$
- (e) $n^2 - 3n + 3$
-

Aufgabe 21 (Komplexität).

Wieviele Gleitpunktoperationen erfordert die Berechnung des Produkts von einer Matrix $A \in \mathbb{R}^{n \times m}$ mit einem Vektor $x \in \mathbb{R}^m$?

- (a) $n + m$
- (b) nm
- (c) $n(2m - 1)$
- (d) $n^2 + m^2$
- (e) m^2
-

Aufgabe 22 (Landau-Symbole).

Seien $(f_n)_{n=1}^{\infty}$ und $(g_n)_{n=1}^{\infty}$ zwei reelle Zahlenfolgen, wobei $f_n = n^2$ und $g_n = n$. Welche der folgenden Aussagen stimmt für $n \rightarrow \infty$ nicht?

- (a) $f_n = O(g_n)$
- (b) $g_n = O(f_n)$
- (c) $f_n = O(n^2)$
- (d) $g_n = O(n^2)$
- (e) $g_n = O(n^{2011})$
-

Aufgabe 23 (Binäre Suche).

Welche der folgenden Aussagen stimmt nicht?

- (a) Die binäre Suche kann falsche Ergebnisse liefern, wenn das Feld nicht aufsteigend sortiert ist.
 - (b) Die binäre Suche kann falsche Ergebnisse liefern, wenn sich das gesuchte Element nicht in dem Feld befindet.
 - (c) Die binäre Suche hat logarithmische Laufzeitkomplexität.
 - (d) Eine Implementierung der binären Suche ist in den Standardbibliotheken verfügbar.
 - (e) Die binäre Suche ist ein Suchalgorithmus.
-

Aufgabe 24 (Rekursive Funktionen).

Was berechnet die folgende Funktion wenn beide Parameter zwischen -100 und 100 liegen?

```
int rec(int m, int n) {  
    int k;  
    if(n==0) return m*m;  
    k=2*m*n;  
    return rec(m+n,0) - k;  
}
```

- (a) 0
 - (b) $m*m$
 - (c) $m + n - 2*m*n$
 - (d) $m*m - 2*m*n$
 - (e) $m*m + n*n$
-

Aufgabe 25 (Nichttriviale Deklarationen).

Was wird hier deklariert?

```
double (*f)(const int, double);
```

- (a) **f** ist ein Zeiger auf eine Funktion, die ein **int** Argument und ein **double** Argument entgegennimmt und ein **double** zurückliefert.
 - (b) **f** ist ein Zeiger auf eine Funktion, die ein unveränderbares **int** Argument und ein **double** Argument entgegennimmt und ein **double** zurückliefert.
 - (c) **f** ist ein Zeiger auf eine Funktion, die ein unveränderbares **int** Argument und ein **double** Argument entgegennimmt und ein Zeiger auf **double** zurückliefert.
 - (d) **f** ist ein Zeiger auf eine Funktion, die entweder ein **int** Argument oder ein **double** Argument entgegennimmt und einen **double** zurückliefert.
 - (e) **f** ist ein Union und kann entweder ein **int** oder ein **double** enthalten.
-