

Nachname: .....

Vorname: .....

MatrNr.: .....

## **Klausur PR2**

HAW-Hamburg, Fakultät Technik und Informatik, Department Informations- und Elektrotechnik

Dr. Robert Heß, 4.7.2008

Bearbeitungsdauer: 90 min

Hilfsmittel: Vorlesungsunterlagen und C/C++ Einführungsbücher (z.B. Erlenkötter)

Jegliche Art von elektronischen Hilfsmitteln sind nicht erlaubt.

### **1. Aufgabe Vektoren (6 Punkte)**

Ein zweidimensionaler Vektor wurde wie folgt definiert:

```
int Feld[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
```

Was ergeben folgende Ausdrücke?

Feld[0][0] .....

Feld[1][2] .....

\*Feld[2] .....

\*\*Feld .....

Feld[3][1] .....

Wie wird die fünf im Feld angesprochen?

.....

### **2. Aufgabe Strukturen (6 Punkte)**

Erstellen Sie eine Struktur für eine Komplexe Zahl mit den Elementen `real` und `imag` vom Typ `double`.

Erstellen Sie zu der eben erstellten Struktur einen neuen Typ mit Namen `tComplex`.

### 3. Aufgabe (14 Punkte)

Erstellen Sie eine Funktion, die Speicher für einen Vektor von `double` reserviert. Die Funktion gibt ggf. vorher den alten Speicher frei und reserviert dann neuen Speicher. Tritt ein Fehler auf, wird eine Fehlermeldung ausgegeben und der Wert -1 zurückgegeben. Bei Erfolg wird eine null zurückgegeben.

```
int getMemory(int size, double **data)
{
    }

}
```

### 4. Aufgabe Programm verstehen (6 Punkte)

```
#include <stdio.h>

unsigned Funktion(int n);

int main()
{
    printf("Funktion(6): %u\n", Funktion(6));

    return 0;
}

unsigned Funktion(int n)
{
    if(n<=2) return 1;
    return Funktion(n-1)+Funktion(n-2);
}
```

Was gibt dieses Programm auf dem Bildschirm aus?

.....

Geben Sie der Funktion einen sinnvollen Namen:

.....

## 5. Aufgabe Verkettete Liste (6 Punkte)

Folgende Deklaration stellt die Struktur für eine doppelt verkettete Liste dar:

```
struct sChain {  
    int number;  
    struct sChain *prev;  
    struct sChain *next;  
};
```

Es seien p1 und p2 die Zeiger auf zwei benachbarte Elemente in der Kette und pn ein Zeiger auf ein neues Element. Erstellen Sie den Programmcode mit dem das neue Element pn hinter das Element p1, bzw. vor das Element p2 eingefügt wird.

## 6. Aufgabe Dateien (6 Punkte)

Mit welchen Funktionen werden in C Dateien geöffnet und geschlossen?

.....

Mit welchen Funktionen werden in C Binärdatei gelesen und geschrieben?

.....

Was wird mit *memory leakage* bezeichnet?

.....

## **7. Aufgabe Fehlersuche** (10 Punkte)

In das Programm auf der folgenden Seite haben sich in 11 Zeilen Fehler eingeschlichen. Tragen Sie die gefundenen Fehler mit dazugehöriger Korrektur in die folgende Tabelle ein. Der erste der 11 Fehler wurde beispielhaft eingetragen.

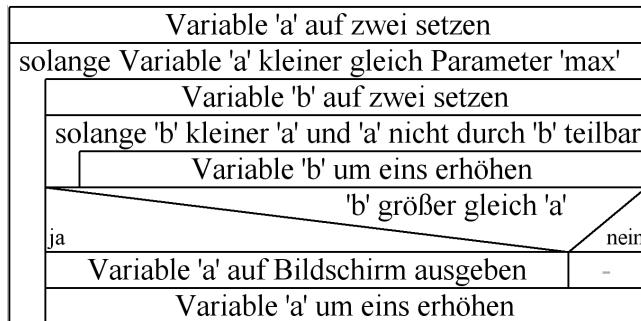
```
/* 1 */ /* Quellcodedatei: Kugel.c von Robert Heß */
/* 2 */ /* Berechnung von Oberfläche und Volumen einer Kugel */
/* 3 */ /* Version 1.00, 24.6.2008 */
/* 4 */
/* 5 */ #include <stdio.h>
/* 6 */
/* 7 */ #define PI 3,141592654 /* Konstante pi für die Berechnungen */
/* 8 */
/* 9 */ void main()
/* 10 */ {
/* 11 */     double r;           /* Radius der Kugel */
/* 12 */     double A;           /* Oberfläche der Kugel */
/* 13 */     double V;           /* Volumen der Kugel */
/* 14 */
/* 15 */     /* Titel ausgeben */
/* 16 */     printf("\n");
/* 17 */     printf("-----+\n");
/* 18 */     printf("| Oberfläche und Volumen einer Kugel |\n");
/* 19 */     printf("-----+\n\n");
/* 20 */
/* 21 */     do
/* 22 */
/* 23 */         /* Radius vom Benutzer abfragen, Ende mit 0 */
/* 24 */         do {
/* 25 */             /* Fragetext ausgeben */
/* 26 */             printf("Geben Sie einen Radius grösser null ein "
/* 27 */                   "(Ende mit 0): ");
/* 28 */             /* Zahl einlesen und auf Gültigkeit prüfen
/* 29 */             if (scanf("%lf", &r) != 1) {
/* 30 */                 printf("Das war keine gültige Zahl!\n");
/* 31 */                 r = -1;
/* 32 */                 fflush(stdin);
/* 33 */                 /* Wenn Radius kleiner null: Fehlermeldung */
/* 34 */                 } else if (r > 0) printf("Der Radius darf nicht kleiner "
/* 35 */                           "als null sein!\n");
/* 36 */             } while (r < 0);
/* 37 */
/* 38 */         /* Oberfläche und Volumen berechnen und ausgeben */
/* 39 */         if (r > 0) {
/* 40 */             A = 4 * PI * r * r;
/* 41 */             V = 4 * PI / 3 * r * r * r;
/* 42 */             printf("Radius der Kugel: %lg\n", r);
/* 43 */             printf("Oberfläche der Kugel: %Lg\n", A);
/* 44 */             printf("Volumen der Kugel: %lg\n\n", V);
/* 45 */         }
/* 46 */
/* 47 */     } while (r == 0); /* wiederholen, bis Radius=0 */
/* 48 */
/* 49 */     /* Schlussmeldung */
/* 50 */     printf("\n")
/* 51 */     printf("Vielen Dank fuer die Benutzung des Programms.\n");
/* 52 */     printf("Einen schönen Tag noch...\n\n");
/* 53 */
/* 54 */     /* Programm mit Rückgabewert null beenden */
/* 55 */     return 0;
/* 56 */ }
```

## 8. Aufgabe Funktion erstellen (14 Punkte)

Eine Funktion wurde wie folgt deklariert:

```
void Funktion(unsigned max);
```

Der Inhalt der Funktion ist durch das folgende Struktogramm wiedergegeben:



Erstellen Sie die Definition der Funktion:

```
void Funktion(unsigned max)
{
```

```
}
```

Welche Ausgabe erzeugt die Funktion, wenn sie mit einem Wert von 12 aufgerufen wird?

Geben Sie der Funktion einen sinnvollen Namen: .....