

Name:

Matrikelnummer:

Bachelorprüfung: Programmieren I

WS09/10

Erlaubte Hilfsmittel: keine

Lösung ist auf den Klausurbögen anzufertigen. (eventuell Rückseiten nehmen)

Bitte legen Sie einen Lichtbildausweis und den Studentenausweis auf den Tisch.

Bearbeitungszeit: 90 Minuten

Unterschrift

Benotung

Aufgabe:	1	2	3	4	5	6		Gesamt	Note
Punkte:	12	16	14	20	12	16		90	
erreicht:									

Name:

Matrikelnummer:

Aufgabe 1 Führen Sie folgendes Programme von Hand aus und notieren Sie die sich ändernden Werte der einzelnen Variablen und geben die Ausgabe an.

a) `Aufgabel1.c`

```
1 #include <stdio.h>
2
3 int f(int i){
4     i = i / 2;
5     switch (i){
6         case 12:
7         case 11:
8         case 10:
9         case 9:
10        case 8: break;
11        case 7:
12        case 6:
13        case 5: return 5;
14        case 4: return i+1;
15        default: return 0;
16    }
17    return i;
18 }
19
20 int main(){
21     int i =12;
22     printf("%d\n", f(i));
23     printf("%d\n", i);
24     return 0;
25 }
26
```

Name:

Matrikelnummer:

b) `Aufgabe1b.c`

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 char* repeat(char c, int i){
5     char* result = (char*)malloc(sizeof(char)*i);
6     for (int j=0;j<i;j++){
7         result[j]=c;
8     }
9     result[4] = '\\0';
10    return result;
11 }
12
13 int main(){
14     char* as = repeat('a',6);
15     printf("%s\\n",as);
16     free(as);
17     return 0;
18 }
```

Name:

Matrikelnummer:

Aufgabe1c.c

```
c) 1 #include <stdio.h>
2 int f(int x){
3     int y=3;
4     for (;y>0;y--){
5         while (x<5){
6             x=x+1;
7             if (x%2==0) continue;
8             printf("x: %i y: %i\n",x,y);
9         }
10    }
11    return y;
12 }
13
14 int main(){
15     printf("%d\n",f(4));
16
17     return 0;
18 }
```

Aufgabeld.c

d)

```
1 #include <stdio.h>
2
3 typedef struct {int x;int y;} Punkt;
4
5 Punkt verschiebe(Punkt* p,int dX,int dY){
6     Punkt result= *p;
7     result.x=p->x+dX;
8     result.y=p->y+dY;
9     return result;
10 }
11
12 void printPunkt(Punkt p){
13     printf("( %i, %i)\n",p.x,p.y);
14 }
15
16 int main(){
17     Punkt p={17,4};
18     printPunkt(p);
19     Punkt p2 = verschiebe(&p,25,38);
20     printPunkt(p);
21     printPunkt(p2);
22     return 0;
23 }
```

Name:

Matrikelnummer:

Aufgabe 2 Schreiben Sie folgende Funktionen auf Zeichenketten:

a) Für Strings ist eine Funktion:

```
boolean containsUpperCase(char* str1);
```

zu implementieren. Es soll zurückgegeben werden, ob ein String einen Großbuchstaben enthält.

b) Für Strings ist eine Funktion:

```
unsigned int equalUpToIndex(char* str1, char* str2)
```

zu implementieren. Es soll berechnet werden, bis zu welchem Index sich die beiden Argumente gleichen.

Für die beiden Strings "hallo du" und "hallo Du" wäre demnach das Ergebnis 4. Es soll also der erste Index sein, an dem sich die beiden String unterscheiden. Fangen zwei String mit unterschiedlichen Zeichen an, so ist demnach das Ergebnis 0.

Name:

Matrikelnummer:

Aufgabe 3 Realisieren Sie die folgenden mathematischen Funktionen in C

a)

$$q(n) = \begin{cases} n, & \text{falls } n/10 = 0, \\ (n\%10) + q(n/10), & \text{sonst} \end{cases}$$

Dabei seien der Operator / die ganzzahlige Division und der Operator % die Modulooperation auf ganzen Zahlen.

Name:

Matrikelnummer:

b)

$$f(m, n) = \begin{cases} n & \text{falls } n^2 = m \\ f(m, n - 1) & \text{falls } m < n^2 \\ f(m, n + 1) & \text{falls } m > n^2 \end{cases}$$

c) Was ist bei der Funktion aus b) als problematisch einzuschätzen.

Name:

Matrikelnummer:

Aufgabe 4 Entwerfen Sie in dieser Aufgabe eine kleine Bibliothek zur Darstellung von Büchern in einem Onlinebookstore. Ein Buchdatensatz besteht dabei aus einem Wert für den Autor, einem Wert für den Titel, einen Wert für die Seitenanzahl und einen Wert für den Preis in Cent.

a) Schreiben Sie eine geeignete Struktur `struct Buchdaten`.

b) Schreiben Sie eine geeignete Konstruktorfunktion, mit der ein Zeiger auf ein Buchdatenobjekt erzeugt und dieses Objekt initialisiert wird.

Name:

Matrikelnummer:

- c) Schreiben Sie eine Funktion, die zwei Buchobjekte als Parameter erhält und einen bool'schen Wert zurückgibt, der anzeigt, ob das erste Buch pro Seite teurer ist als das zweite.

- d) Schreiben Sie eine Funktion, die als ersten Parameter ein Buchobjekt erhält und als zweiten Parameter eine ganze Zahl. Der zweite Parameter gibt die Prozentzahl an, um die der Preis des Buches erhöht werden soll.

Hinweis Das Objekt wird verändert!

Name:

Matrikelnummer:

Aufgabe 5 Gegeben Sei folgende Implementierung von einfach verketteten Listen, wie aus der Vorlesung bekannt. Die Listenelemente sind ganze Zahlen

```
1 typedef enum {false,true} boolean;
2
3 struct Liste{
4     boolean isEmpty;
5     int head;
6     struct Liste* tail;
7 };
8
9 typedef struct Liste Li;
10 Li* newLi(char head,Li* tail);
11 Li* newEmptyLi();
```

Implementieren Sie folgende Funktionen rekursiv:

a) `boolean containsNumberGreaterThan(Li* this, int i);`

Die Funktion soll genau dann `true` ergeben, wenn in der Liste `this` eine Zahl enthalten ist, die echt größer als der Parameter `i` ist.

Name:

Matrikelnummer:

Aufgabe 6 Erklären Sie in kurzen Worten.

a) In was unterscheiden sich die beiden Operatoren `&` und `&&`?

b) Was bewirkt der Befehl `continue`?

Name:

Matrikelnummer:

- c) Funktionen können in C als Parameter übergeben werden. In wiefern ist das nützlich, wenn man einen allgemeinen Sortieralgorithmus implementiert?

- d) Die Indexschreibweise beim Elementezugriff in Arrays versteckt eine Addition auf Zeigern. Wie kann man einen Zugriff `ar[i]` auch direkt mit einem Zeigerzugriff schreiben?