
Aufgaben zur Klausur **C** im WS 2001/02 (IA 302)

Zeit: 60 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Sollten Unklarheiten oder Mehrdeutigkeiten bei der Aufgabenstellung auftreten, so notieren Sie bitte, wie Sie die Aufgabe interpretiert haben.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 6 Seiten

Aufgabe 1:

Gegeben sei das folgende Programm

```
#include <stdio.h>

int x1 [] = {1,2,3};
int x2 [] = {4,2,6};
int x3 [] = {9,7,5};

int * a[] = {x3,x2,x1};
int ** pp, *p, i;

int main(void) {

    pp = a, i = * * pp;
    printf("%d\n",i);
    p = *(++pp), i = *(++p);
    printf("%d\n",i);
    i = *( *(a + 1 ) +1 );
    printf("%d\n",i);
    pp = a + 1, i = pp[1][2];
    printf("%d\n",i);
    pp = a + 1, p = pp[-1] + 1; i = p[-1];
    printf("%d\n",i);
    pp = a + 2, --pp, i = (*pp == a[1]);
    printf("%d\n",i);
    p = a[2] + 1, i = *p;
    printf("%d\n",i);
    pp = 1 + a, p = 2 + *pp, i = *(1 + p);
    printf("%d\n",i);
    return 0;
}
```

Wird in diesem Programm an irgend einer Stelle eine implizite Konversion durchgeführt?

ja nein

Begründung:

.....

Wie sieht die Ausgabe des Programms aus ?

1)

2)

3)

4)

5)

6)

7)

8)



Aufgabe 2:

Gegeben sei das folgende C-Programm:

```
#include <stdio.h>

int h(int i) {
    printf("h");
    return i < 10 ? i : h( (i % 10) + (i / 10));
}

int g(int i) {
    printf("g");
    return ( (i % 10) == 7) || (i ≥ 10 && g(i / 10));
}

int f(int i) {
    printf("f");
    return (i % 7 == 0) || g(i) || (h(i) == 7);
}

int main() {
    printf(" %d\n",f(77));
    printf(" %d\n",f(61));
    printf(" %d\n",f(99));
    printf(" %d\n",f(7002));

    return 0;
}
```

Welche vier Ausgabezeilen erzeugt dieses Programm:

- 1)
- 2)
- 3)
- 4)

Aufgabe 3:

Gegeben ist das folgende C Programmstück. Dieses soll in eine header Datei *p.h* und eine Implementierungsdatei *p.c* aufgeteilt werden, so daß es von mehreren anderen Modulen verwendet werden kann. Kennzeichnen Sie durch Ankreuzen des *.h* Feldes für die header Datei oder *.c* für die Implementierungsdatei, in welche Datei die einzelnen Codestücke übertragen werden müssen.

<code>static int</code> zwischenErgebnis;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>#include <stdio.h></code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>typedef unsigned long</code> Nat0;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>static int</code> zwischenErgebnis;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>const</code> Nat0 MAX = 8888;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>extern</code> Nat0 ergebnis;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>typedef struct</code> Rec * PtrRec;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
Nat0 ergebnis = 42;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>static int</code> zwischenErgebnis = 4711;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>static int</code> f(PtrRec x1);	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>int</code> g(PtrRec x1);	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>#define</code> MIN 1111	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>struct</code> Rec { Nat0 d; Nat0 s; } Rec;	<input type="checkbox"/> .h	<input type="checkbox"/> .c
Nat0 toNat0(struct Rec);	<input type="checkbox"/> .h	<input type="checkbox"/> .c
void outRec(FILE * out, PtrRec t);	<input type="checkbox"/> .h	<input type="checkbox"/> .c
Nat0 toNat0(struct Rec t) { zwischenErgebnis = t.d - t.s; return t.d + t.s; }	<input type="checkbox"/> .h	<input type="checkbox"/> .c

<code>int check (PtrRec t) { return t→d ≥ MIN; }</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>void outRec(FILE * out, PtrRec t) { fprintf(out, "%1u,%1u\n", t→d, t→s); }</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>extern Nat0 einPaarZahlen[];</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>int g(PtrRec x1) { return f(x1) + 2; }</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>#define initRec(x) ((x).d = 0,(x).s = 0)</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>struct Array { int d[MIN]; };</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>extern struct Array einArray;</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
<code>static int f(PtrRec x1) { return toNat0(*x1); }</code>	<input type="checkbox"/> .h	<input type="checkbox"/> .c
