
Aufgaben zur Klausur C im SS 2000 (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 5 Seiten

Aufgabe 1:

Gegeben sei das folgende Programm:

```
#include <stdio.h>

char * tab [] = { "Falle" , "Maus" , "Kaefig" , "Gericht" };

char ** ptab [] = { tab + 3, tab + 2, tab + 1, tab };

char *** ppp = ptab;

int main ( int argc , char * argv [] )
{
    printf( "%s\n" , * ( * ( ppp + 2 ) - 1 ) + 1 );
    printf( "%s\n" , ppp [2] [0] + 3 );
    printf( "%s\n" , * ( * ++ppp + 1 ) + 2 );
    printf( "%s\n" , * * ppp + 4 );

    return 0;
}
```

Welche Ausgabezeilen liefert dieses Programm:

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

Wieviel Speicher wird von den Variablen tab, ptab und ppp und den in den Initialisierungen vorkommenden Konstanten benötigt? Geben Sie hierfür einen Ausdruck mit dem **sizeof**-Operator an.

.....
.....

Aufgabe 2:

Gegeben sei das folgende Programm:

```
#include <ctype.h>
#include <stdio.h>

typedef char (*Translate)(char);

char ident(char c) {
    return c;
}

char toUpper(char c) {
    return islower(c) ? c - ('a' - 'A') : c;
}

char toLower(char c) {
    return isupper(c) ? c + ('a' - 'A') : c;
}

char toAlnumSpace(char c) {
    return isalnum(c) ? c : ' ';
}

char whiteSpaceToSpace(char c) {
    return isspace(c) ? ' ' : c;
}

char cntrlToSpace(char c) {
    return iscntrl(c) ? ' ' : c;
}

char spaceToMinus(char c) {
    return c == ' ' ? '-' : c;
}

char * translateString(char* d, char * s, Translate tf) {
    char * d1 = d;

    while (*s) {
        *d1++ = tf(*s++);
    }

    *d1 = 0;
    return d;
}
```

```

char * translateString2(char* d, char * s, Translate * ptf) {
    char * d1 = d;
    char ch;

    while ((ch = *s++)) {

        Translate * ptf1 = ptf;
        Translate tf;

        while ((tf = *ptf1++)) {
            ch = tf(ch);
        }
        *d1++ = ch;
    }
    *d1 = 0;
    return d;
}

char * tab[] = {
    "-abc-123-",
    "+\n+\n+\n+\n",
    "123 456",
    "a+-+-+z"
};

Translate t1[] = {whiteSpaceToSpace, spaceToMinus, (Translate)0};
Translate t2[] = {whiteSpaceToSpace, toUpper, (Translate)0};

#define OUT printf("%s\n", res)

int main(int argc, char* argv[]) {
    char res[256];

    translateString(res, tab[0], ident); OUT;

    translateString(res, tab[3], toUpper); OUT;

    translateString(res, tab[1], whiteSpaceToSpace);
    translateString(res, res, spaceToMinus); OUT;

    translateString2(res, tab[0], t2); OUT;

    translateString2(res, tab[1], t1); OUT;

    return 0;
}

```

Welche Ausgabezeilen liefert dieses Programm:

1)

2)

3)

4)

5)

