

Im folgenden ist der Quelltext der Vorbereitungsaufgaben zu finden. Das Programm erwartet vier Parameter, drei Strings und eine Integer-Zahl. Dabei wird von dem ersten Parameter die Länge gemessen mit `mystrlen` und auch wird dieser für `mystrcpy` verwendet. Zusätzlich wird von `mystrcat` der zweite Parameter benötigt. Hier wird nämlich der zweite noch an die erste Zeichenkette angehängen. Die letzten beiden Parameter sind für das struct da, und zwar wird einmal ein Name eingelesen und ein Alter.

Ein Beispielaufruf könnte so aussehen:

```
> ./vorbereitung01 hallo du Ich 100
> Länge von hallo ist 5
> Strcpy von 0xbffff9e5 nach 0xbffff90c: hallo
> Strcat von hallo und du ergibt: hallodu
> Person vorbelegt: Name: Heinz Alter: 1
> initPerson: Name: Ich Alter: 100
```

Wenn jetzt z.B. der Name länger als 20 Zeichen ist, wird der vorbelegte Eintrag nicht verändert:

```
> ./vorbereitung01 hallo du IchBinSehrLangBlaBlaBla 100
> Länge von hallo ist 5
> Strcpy von 0xbffff9bd nach 0xbffff8dc: hallo
> Strcat von hallo und du ergibt: hallodu?
> Person vorbelegt: Name: Heinz Alter: 1
> initPerson: Name: Heinz Alter: 1
```

```
1 #include <stdio.h>
2
3 #define NAMELENGTH 20
4
5 struct Person
6 {
7     char name[NAMELENGTH];
8     int alter;
9 };
10
11 int mystrlen(const char* str) {
12     int i;
13     int len = 0;
14     for(i=0; str[i] != '\0'; i++) {
15         len++;
16     }
17     return len;
18 }
19
20 char* mystrcpy(char* s1, const char* s2) {
21     int i;
```

```
22     for(i=0;s2[i] != '\0';i++) {
23         s1[i] = s2[i];
24     }
25     while(s1[i] != '\0') {
26         s1[i] = '\0';
27     }
28     return s1;
29 }
30
31 char* mystrcat(char* s1, const char* s2) {
32     int j;
33     int len = mystrlen(s1);
34     for(j=0;s2[j] != '\0';j++) {
35         s1[len] = s2[j];
36         len++;
37     }
38     return s1;
39 }
40
41 struct Person* initPerson(struct Person* p, const char* name,
42     int alter) {
43     if(mystrlen(name) <= NAMELENGTH) {
44         mystrcpy(p->name,name);
45         p->alter = alter;
46         return p;
47     }
48     else
49         return 0;
50 }
51 /**
52  * Erwartet vier Parameter auf der Konsole, und zwar s1 s2 fuer
53  *   mystrcpy und mystrcat
54  * und s3 und s4 fuer den Namen des struct Person und das Alter.
55  * So kann auch eine Zeichenkette >20 getestet werden.
56  */
57 int main(int argc, char* argv[]) {
58     // da die Parameter veroendert werden, werden sie hier
59     // fuer weiteres arbeiten kopiert
60     char copy[20], catS1[20], catS2[20];
61     mystrcpy(copy,argv[1]);
62     mystrcpy(catS1,argv[1]);
63     mystrcpy(catS2,argv[2]);
64
65     // Testen von mystrlen, mystrcpy und mystrcat
```

```
65     printf("Laenge_von_%s_ist_%i\n",argv[1],mystrlen(argv
        [1]));
66     printf("Strcpy_von_%p_nach_%p:_%s\n",argv[1],copy,
        mystrcpy(copy,argv[1]));
67     printf("Strcat_von_%s_und_%s_ergibt:_%s\n",argv[1],argv
        [2],mystrcat(catS1,catS2));
68
69     // instanziert struct Person und belegt es vor
70     struct Person p;
71     mystrcpy(p.name,"Heinz");
72     p.alter = 1;
73     printf("Person_vorbelegt: _Name:_%s_Alter:_%d\n",p.name,
        p.alter);
74
75     // initPerson wird getestet
76     initPerson(&p,argv[3],atoi(argv[4]));
77     printf("initPerson: _Name:_%s_Alter:_%d\n",p.name,p.
        alter);
78
79     return 0;
80 }
```
