



Embedded Systeme 1 & 2

Klassen T-2/I-2 // 2018-2019

a.08 - C - Funktionszeiger

Übung 1

Entwickeln Sie ein kleines Programm, das den Aufruf eines beliebigen Satzes von Befehlen mit ihren Argumenten ab einer Zeichenkette erlaubt. Das Programm muss allgemein gehalten sein und Funktionszeiger verwenden.

Das Programm muss zum Beispiel für die nachstehenden Befehlszeilen korrekt funktionieren:

```
1) "command1 10 20 30"  
2) "command2 10 foo_bar"  
3) "command3 30 40.20 50e-2"  
4) "command4"
```

Übung 2

Gegeben seien die folgenden 4 Funktionen:

```
1) int funca (RET *ptr, int max) ;  
2) int funcb (RET *ptk, int min) ;  
3) int funcc (RET *trd, int avg) ;  
4) int funcd (RET *utr, int value) ;
```

- 1) Deklarieren Sie eine Funktionszeiger-Tabelle ("array"), die diese Funktionen speichern kann, und initialisieren Sie anschließend die Tabelle mit diesen Werten.
- 2) Rufen Sie die Funktionen mithilfe der zuvor initialisierten Tabelle auf.



Übung 3

Welche Werte werden nach der Ausführung des folgenden Codes in der Tabelle "out" enthalten sein?

```
#define ARRAY_SIZE(x) (sizeof(x)/sizeof(x[0]))

typedef int (*fnct_t) (int);

static int f1 (int a) { return (25 + (3 * a)); }
static int f2 (int a) { return (192 - a); }
static int f3 (int a) { return (a - 100); }
static int f4 (int a) { return (a % 8); }

static fnct_t tabf[] = {f1, f2, f3, f4};

static int fnct(int x) {
    unsigned k = (x >> 6) % ARRAY_SIZE(tabf);
    return tabf[k](x);
}

int main() {
    int in[] = {10, 90, 120, 160, 180, 250};
    int out[ARRAY_SIZE(in)];

    for(unsigned i = 0; i<ARRAY_SIZE(in); i++) {
        out[i] = fnct(in[i]);
    }

    return 0;
}
```

i	fnct	a	out[i]
0			
1			
2			
3			
4			
5			



Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg