

7.9.A Projekte/Logging-Modul – Arbeitsblatt

1 Situation

Das Modul `ferrou.c` soll drei Funktionen zur Verfügung stellen:

- `ferropen(dateiname)` öffnet eine Datei. Zweck: Ausgabe von Fehlermeldungen.
- `ferrou("beispiel.c", 14, "Fehleingabe!")` schreibt eine Zeile in die geöffnete Datei, z.B.:

```
1 2015.02.25,10:03:20:beispiel.c,14:Fehleingabe!
```

Der erste Parameter (hier `"beispiel.c"`) ist dazu gedacht, dass man den Namen der Quelltextdatei angibt. Der zweite Parameter (hier 14) soll angeben, in welcher Zeile der Fehler bemerkt wurde. Im dritten Parameter (hier `"Fehleingabe!"`) kann der Programmierer, der das Modul benutzt, die Art des Fehlers angeben.

Am Beginn der Ausgabezeile soll der Zeitpunkt stehen, zu dem der Fehler aufgetreten ist.

- `ferrclose()` schließt die Datei.

Diese Funktionen sollen es einem Programmierer erleichtern, die immer gleiche Ausgabe von Fehlermeldungen in eine Log-Datei zu vereinfachen. Ein vollständiges Anwendungsbeispiel ist hier:

```
1 #include <stdio.h>
2 #include "ferrou.h"
3 int main(void)
4 {
5     int rc;
6     double r1, r2, rp;
7     rc=ferropen("beispiel.log"); /* Log-Datei wird geoeffnet */
8     if(rc!=0) return -1; /* schlug fehl */
9     printf("Dieses_Programm_berechnet_R1||R2.\n");
10    printf("Eingabe_R1:"); scanf("%lf", &r1);
11    printf("Eingabe_R2:"); scanf("%lf", &r2);
12    if(r1==r2)
13        ferrout(__FILE__, __LINE__, "Fehleingabe!"); /* Log-Datei wird geschrieben */
14    else
15    {
16        rp=r1*r2/(r1+r2);
17        printf("Ergebnis:%lf\n", rp);
18    }
19    ferrclose(); /* Log-Datei wird geschlossen */
20    return 0;
21 }
22 }
```

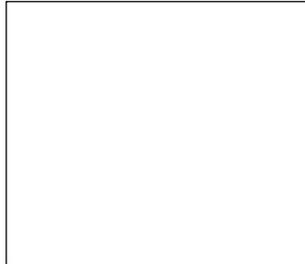
2 Aufgaben

- Die Funktionen sollen erstellt werden.
- Zuerst soll das Modul durch eine geeignete `main()`-Funktion in diesem Modul getestet werden.
- Anschließend soll das Modul `ferrou.c` von dem eigentlichen Nutzprogramm getrennt benutzt werden können.

ferrout.h



ferrout.c



beispiel.c

```
#include "ferrout.h"

int main(void)
{

}

}
```

Compilieren: gcc ferrout.c beispiel.c

3 Hinweise

- In diesem Modul ist eine (Modul-) globale Variable `FILE* f` sinnvoll.
- Es empfiehlt sich, das folgende Makro anzulegen:

```
1 #define ferr(meintext) ferrout(__FILE__, __LINE__, meintext)
```

Damit vereinfacht sich der Aufruf im Beispielprogramm zu `ferr("Fehleingabe!");`.