

## Angewandte Informatik – Probeklausur – 22. Oktober 2015

Prof. Dr. Peter Gerwinski, Wintersemester 2015/16

|               |  |
|---------------|--|
| Name:         |  |
| Matrikel-Nr.: |  |

Zeit: 120 Minuten

Zulässige Hilfsmittel:

- Schreibgerät
- Beliebige Unterlagen in Papierform und/oder auf Datenträgern
- Elektronische Rechner (Notebook, Taschenrechner o. ä.)  
*ohne* Zugang zu Datennetzen jeglicher Art

Nur die o. a. zulässigen Hilfsmittel dürfen sich während der Klausur im Arbeitsbereich befinden. WLAN-, Bluetooth- und sonstige Funkeinheiten von Notebooks o. ä. sind auszuschalten; ggf. dafür vorhandene physische Schalter sind zu benutzen. Mobiltelefone, Geräte mit mobilem Internet-Zugang u. ä. sind auszuschalten und in der Tasche zu verstauen.

Die reguläre Maximalpunktzahl beträgt 42 Punkte.  
Bei besonderen Leistungen sind Zusatzpunkte möglich.  
Mit 20 erreichten Punkten gilt die Klausur als bestanden.

Die Beispielprogramme können Sie sich unter  
<https://www.cvh-server.de/~pgerwinski/ainf-probeklausur-20151022/>  
herunterladen.

Lösungen in digitaler Form können Sie per E-Mail an [peter.gerwinski@hs-bochum.de](mailto:peter.gerwinski@hs-bochum.de) einreichen. Das Betreff sollte dabei die folgende Zeichenfolge enthalten: **aznj93pd**

## Aufgabe 1: Länge von Strings

Strings werden in der Programmiersprache C durch Zeiger auf **char**-Variable realisiert.

Beispiel: **char** \*hello\_world = "Hello,\_world!\n"

Die Systembibliothek stellt eine Funktion `strlen()` zur Ermittlung der Länge von Strings zur Verfügung (`#include <string.h>`).

- (a) Auf welche Weise ist die Länge eines Strings gekennzeichnet? (1 Punkt)

- (b) Wie lang ist die Beispiel-String-Konstante "Hello, \_world!\n", und wieviel Speicherplatz belegt sie? (2 Punkte)

- (c) Schreiben Sie eine eigene Funktion `int strlen (char *s)`, die die Länge eines Strings zurückgibt. (4 Punkte)

Wir betrachten nun die folgenden Funktionen:

```
int fun_1 (char *s)
{
    int x = 0;
    for (int i = 0; i < strlen (s); i++)
        x += S[i];
    return x;
}
```

```
int fun_2 (char *s)
{
    int i = 0, x = 0;
    int len = strlen (s);
    while (i < len)
        x += s[i++];
    return x;
}
```

(d) Was bewirken die beiden Funktionen? (2 Punkte)

(e) Wie oft greifen die beiden Funktionen auf die einzelnen Zeichen `s[i]` des String `s` zu? Welche der beiden Funktionen arbeitet schneller? (Sie dürfen für die Funktion `strlen()` Ihre eigene Version voraussetzen.) (4 Punkte)

(f) Schreiben Sie eine eigene Funktion, die dieselbe Aufgabe erledigt wie `fun_2()`, nur effizienter. (4 Punkte)



(b) Wofür stehen die Zahlen? (2 Punkte)

(c) Ergänzen Sie das Programm derart, daß seine `main()`-Funktion `int main (void)` lautet und eine `return`-Anweisung hat, wobei die in Aufgabenteil (a) festgestellte Eigenschaft erhalten bleiben soll. (5 Punkte)

#### Aufgabe 4: Ausgabe von Hexadezimalzahlen

Schreiben Sie eine Funktion `void print_hex (uint32_t x)`, die eine gegebene vorzeichenlose 32-Bit-Ganzzahl `x` als Hexadezimalzahl ausgibt. (Der Datentyp `uint32_t` ist mit `#include <stdint.h>` verfügbar.)

Verwenden Sie dafür *nicht* `printf()` mit der Formatspezifikation `%x` als fertige Lösung, sondern programmieren Sie die nötige Ausgabe selbst.

Die Verwendung von `printf()` mit anderen Formatspezifikationen wie z. B. `%d` oder `%c` oder `%s` ist hingegen zulässig.

(8 Punkte)

Abgabe auf Datenträger ist erlaubt, aber nicht zwingend.

Raum für Notizen:

*Viel Erfolg!*