

9. Übung

Ausgabe Abgabe
11.12.15 8.1.16

Bitte bei der Abgabe Name der Mitglieder einer Gruppe, Nummer der Übung/Teilaufgabe und Datum auf den Lösungsblättern nicht vergessen! Darauf achten, dass die Lösungen beim richtigen Tutor abgegeben werden. Achten Sie bei Programmieraufgaben außerdem darauf, dass diese im Linuxpool kompilierbar sind. Nutzen Sie dazu die Flags `-std=c99`, `-Wall` und `-pedantic`. Es sollten keine Warnungen auftauchen.

Zu spät abgegebene Lösungen werden nicht berücksichtigt!

Aufgabe 1: Betriebssysteme (1 Punkte)

Mit dieser Aufgabe soll der erste Teil der Veranstaltung abgeschlossen werden. Befassen Sie sich mit den bisherigen Themen und betrachten Sie diese Aufgabe auch als Teil der Klausurvorbereitung.

- (a) Fertigen Sie zum Thema Betriebssysteme eine Mindmap mit mindestens 30 Begriffen an. In dieser Mindmap sollten alle Themengebiete vertreten sein.
- (b) Nennen Sie 10 Begriffe, die sie gut verstanden haben und erklären können.
- (c) Nennen Sie 10 Begriffe, die sich nicht verstanden haben und diskutieren möchten.

Aufgabe 2: Wireshark (1 Punkte)

Laden sie das Programm Wireshark herunter (<http://www.wireshark.org>) und installieren Sie es. Machen Sie sich mit der Benutzung von Wireshark vertraut! Stellen Sie eine Verbindung mit der Webseite <http://www.inf.fu-berlin.de> und schneiden Sie den gesamten Verbindungsaufbau mit Wireshark mit.

- (a) Wie nennt man Werkzeuge wie Wireshark? Welche anderen gibt es noch?
- (b) Welche Protokolle sind beteiligt?
- (c) Beschreiben Sie kurz die Aufgabe dieser Protokolle.

Aufgabe 3: Sockets (3 Punkte)

In dieser Aufgabe sollen Sie eine einfache Client/Server-Anwendung mit Internet sockets (TCP) implementieren. Der Client/Server sollte wie folgt gestartet werden:

```
./server <port>  
./client <port>
```

- (a) Implementieren Sie einen Server, der auf Anfragen einer weihnachtlichen Zeile die darauf folgende Zeile zurücksendet. Wenn der Server beispielsweise die Nachricht "Jingle bells, jingle bells," empfängt, soll er "Jingle all the way!" zurücksenden.

Beachten Sie folgende Punkte:

- Es reicht aus, wenn der Server nur gleichzeitig eine Verbindung bearbeiten kann.
- Es sollte mindestens fünf Zeilen zur Verfügung stehen.
- Überlegen Sie sich ein Verfahren für unbekannte Anfragen, d.h. eine empfangende Zeile ist unbekannt.
- Ein Client soll die Verbindung wieder trennen können und der Server soll danach wieder zur Verfügung stehen. Implementieren Sie dazu ein geeignetes Verfahren.
- Überlegen Sie sich, wie die Zeilen verwaltet werden sollten.

Hinweis:

Verwenden Sie Ihnen bekannte Datenstrukturen aus anderen Vorlesungen. Sie dürfen vorhandenen Implementierungen und Bibliotheken für die Datenstruktur verwenden.

- (b) Implementieren Sie einen Client, der eine Verbindung zum Server herstellt, Anfragen vom Benutzer über die Konsole an den Server weiterleitet und die Antwort auf der Konsole ausgibt. Bei Eingabe von "Stopp" soll die Verbindung zum Server beendet werden.

- (c) *Bonusaufgabe für einen weiteren Programmierpunkt:*

Erweitern Sie Ihren Server und Client um die Ausgabe von ASCII-Bildern. Der Server soll auf Anforderung vom Benutzer via Client ein weihnachtliches ASCII-Bild an den Client senden. Der Client soll dieses Bild dann auf der Konsole ausgeben. Für das ASCII-Bild können Sie Beispiele aus dem Internet verwenden, aber geben Sie die Quelle an. Beispiel:

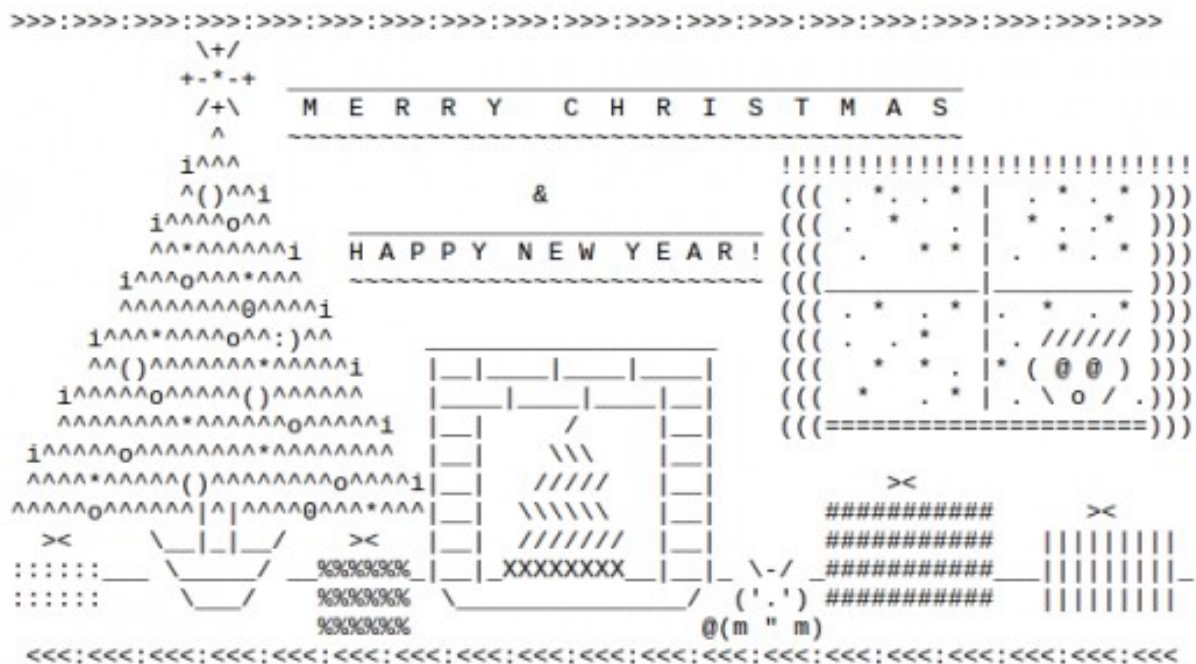


Abbildung 1: Quelle:http://s2.hubimg.com/u/5880097_f520.jpg , 9.12.2015

Hinweis:

Ein guter Einstieg in die Socketprogrammierung ist *Beej's Guide to Network Programming - Using Internet Sockets* frei verfügbar auf <http://beej.us/guide/bgnet/>