U1 1. Übung

U1-1 UNIX/Linux Benutzerumgebung

- Spezielle Projektverzeichnisse zur Bearbeitung der Übungsaufgaben
  - `/proj/i4gdi/`*login-Name*
  - Wechseln des Arbeitsverzeichnisses dorthin (cd = change directory):
    `cd /proj/i4gdi/`*login-Name*
  - Einrichten eines Verzeichnisses für Aufgabe0:
    `mkdir aufgabe0`
  - Anzeigen der Dateien in einem Verzeichnis:
    aktuelles Verzeichnis anzeigen: `ls` oder `ls -l`
    Verzeichnis `aufgabe1` anzeigen: `ls -l aufgabe1`
- Kommandointerpreter (Shell)
  - Programm, das Kommandos entgegennimmt und ausführt
  - verschiedene Varianten, am häufigsten unter Linux: bash oder tcsh

U1-1 UNIX/Linux Benutzerumgebung (2)

- Sonderzeichen
  - einige Zeichen haben unter UNIX besondere Bedeutung
  - Funktionen:
    - Korrektur von Tippfehlern
    - Steuerung der Bildschirm-Ausgabe
    - Einwirkung auf den Ablauf von Programmen
- Übersicht: (<CTRL> = <STRG>)

<BACKSPACE>	letztes Zeichen löschen (manchmal auch **<DELETE>**)
<CTRL> U	alle Zeichen der Zeile löschen
<CTRL>C	Interrupt - Programm wird abgebrochen
<CTRL>Z	Stop - Programm wird gestoppt (mit `fg` läuft's weiter)
<CTRL>D	End-of-File
<CTRL>S	Ausgabe am Bildschirm wird angehalten
<CTRL>Q	Ausgabe am Bildschirm läuft weiter

U1-1 UNIX/Linux Benutzerumgebung (3)

- Editor
  - verschiedene Editoren unter UNIX verfügbar
    - vi
    - emacs
  - für unerfahrene Benutzer zu empfehlen: **`kate`**
  - Starten
    - durch Eingabe von `kate` in einer Shell
    - oder über Auswahlmenü von KDE

U1-1 UNIX/Linux Benutzerumgebung (4)

- Manual: man-pages
  - Aufgeteilt nach verschiedenen *Sections*

  (1) Kommandos
  (2) Systemaufrufe
  (3) Bibliotheksfunktionen
  (5) Dateiformate (spezielle Datenstrukturen, etc.)
  (7) verschiedenes (z.B. Terminaltreiber, IP, …)
- man-Pages werden normalerweise mit der Section zitiert: **`printf(3)`**
- Aufruf unter Linux

```
man [section] Begriff

z.B. man 3 printf
```

- Suche nach Sections: **`man -f Begriff`**
  Suche von man-Pages zu einem Stichwort: **`man -k Stichwort`**

U1-2 Erläuterung zu Aufgabe 0 und 1

- spezielle Aufrufoptionen des Compilers

`gcc -pedantic`	liefert Warnungen in allen Fällen, die nicht 100% dem ANSI-C-Standard entsprechen
... -Wall	liefert in vielen evtl. zweifelhaften Situationen (die zwar korrekt sein könnten, aber häufig nicht sind) Warnungen

  diese Optionen führen zwar oft zu nervenden Warnungen, helfen aber auch dabei, Fehler schnell zu erkennen:

  Empfehlung: `gcc -pedantic -Wall -o aufgabe1 aufgabe1.c`

- Beispiel zur Verwendung von getchar() siehe Vorlesungsfolien

- Um getchar verwenden zu können, muss eine spezielle include-Datei eingebunden werden

```
#include <stdio.h>
```

  am Anfang des Programms eingeben