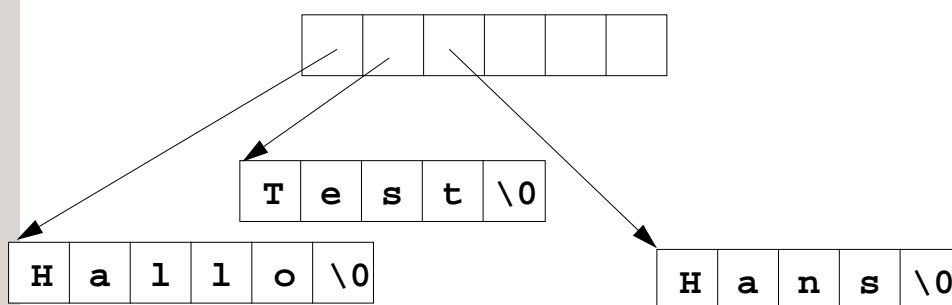


# 3. Tafelübung

- Lösung der wsort-Aufgabe
- Header-Files, Typedefs, ...

## wsort - Datenstrukturen (1)

- Array von Zeigern auf Zeichenketten



- Vorteile:
  - ◆ schnelles Sortieren, da nur Zeiger vertauscht werden müssen
  - ◆ Zeichenketten können beliebig lang sein

## wsort - Datenstrukturen (2)

### ■ Array von Zeichenketten

H	a	l	l	o	\0	\0	\0	T	e	s	t	\0	\0	\0	\0	H	a	n	s	...
---	---	---	---	---	----	----	----	---	---	---	---	----	----	----	----	---	---	---	---	-----

### ■ Vorteile:

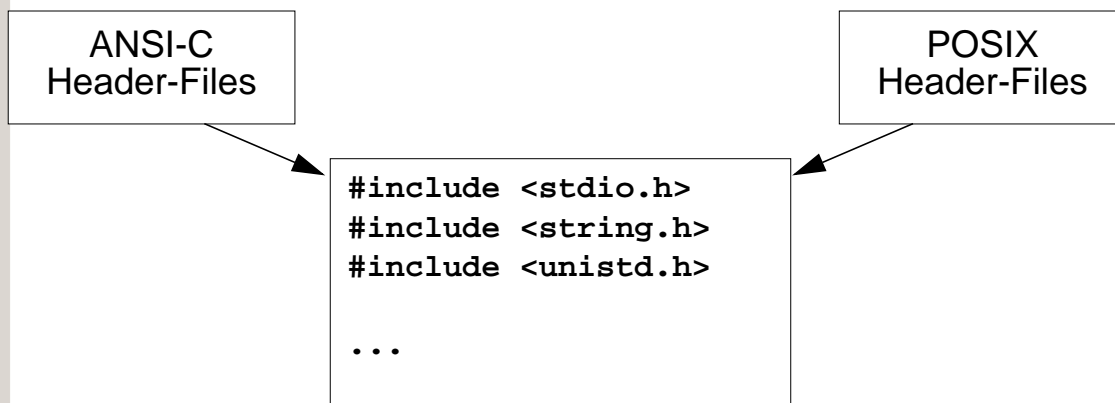
- ◆ einfach
- ◆ strcmp als Vergleichsfunktion verwendbar

### ■ Nachteile:

- ◆ hoher Kopieraufwand
- ◆ Maximale Länge der Worte muß bekannt sein

## Header-Files: ANSI und POSIX

- In den Standards ANSI-C und POSIX.1 sind Header-Files definiert, mit
  - ◆ Funktionsprototypen
  - ◆ typedefs
  - ◆ Makros und Defines
  - ◆ Wenn in der Aufgabenstellung nicht anders angegeben, sollen ausschließlich diese Header-Files verwendet werden.



# ANSI-C Header-Files

- **assert.h**: assert()-Makro
- **ctype.h**: Makros und Funktionen für Characters (z.B. tolower(), isalpha())
- **errno.h**: Fehlerauswertung (z.B. errno-Variable)
- **float.h**: Makros für Fließkommazahlen
- **limits.h**: Enthält Definitionen für Systemschranken
- **locale.h**: Funktion setlocale()
- **math.h**: Mathematische Funktionen für double
- **setjmp.h**: Funktionen setjmp(), longjmp()
- **signal.h**: Signalbehandlung
- **stdarg.h**: Funktionen und Makros für variable Argumentlisten
- **stddef.h**: Def. von ptrdiff\_t, NULL, size\_t, wchar\_t, offsetof, errno
- **stdio.h**: I/O Funktionen (z.B. printf(), scanf(), fgets())
- **stdlib.h**: Hilfsfunktionen (z.B. malloc(), getenv(), rand())
- **string.h**: Stringmanipulation (z.B. strcpy())
- **time.h**: Zeitmanipulation (z.B. time(), ctime(), strftime())

# POSIX Header-Files

- **dirent.h**: opendir(), readdir(), rewinddir(), closedir()
- **fcntl.h**: open(), creat(), fcntl()
- **grp.h**: getgrgid(), getgrnam()
- **pwd.h**: getpwuid(), getpwnam()
- **setjmp.h**: sigsetjmp(), siglongjmp()
- **signal.h**: kill(), sigemptyset(), sigfillset(), sigaddset(), sigdelset(), sigismember(), sigaction, sigprocmask(), sigpending(), sigsuspend()
- **stdio.h**: ctermid(), fileno(), fdopen()
- **sys/stat.h**: umask(), mkdir(), mkfifo(), stat(), fstat(), chmod()
- **sys/times.h**: times()
- **sys/types.h**: enthält betriebssystemabhängige Typdefinitionen
- **sys/utsname.h**: uname()
- **sys/wait.h**: wait(), waitpid()
- **termios.h**: cfgetospeed(), cfsetospeed(), cfgetispeed(), cfsetispeed(), tcgetattr(), tcsetattr(), tcsendbreak(), tcdrain(), tcflush(), tcflow()
- **time.h**: time(), tzset()
- **utime.h**: utime()
- **unistd.h**: alle POSIX-Funktionen, die nicht in den obigen Header-Files definiert sind (z.B. fork(), read())

# POSIX Datentypen

---

## ■ Betriebssystemabhängige Typen aus `<sys/types.h>`:

- `dev_t`: Gerätenummer
- `gid_t`: Gruppen-ID
- `ino_t`: Seriennummer von Dateien (Inodenummer)
- `mode_t`: Dateiattribute (Typ, Zugriffsrechte)
- `nlink_t`: Hardlink-Zähler
- `off_t`: Dateigrößen
- `pid_t`: Prozeß-ID
- `size_t`: entspricht dem ANSI-C `size_t`
- `ssize_t`: Anzahl von Bytes oder -1
- `uid_t`: User-ID