

### Beispiel 1: Doppelt verkettete Liste (12 Punkte)

Entwickeln Sie ein Programm, das eine doppelt verkettete Liste von Zeichenketten verwaltet. Verwenden Sie dazu die folgende Datenstruktur:

```
struct List {
    char *text;
    struct List *prev;
    struct List *next;
};
```

Entwerfen Sie folgende Funktionen:

```
int AddElem(struct List **list, char *c);
// Einfügen eines neuen Strings am Anfang der Liste
int DelElem(struct List **list, char *c);
// Suchen und Löschen eines Elements aus der Liste
int RevertList(struct List **list);
// Umdrehen der Listenverkettung (1<->n, 2<->n-1, ...)
int PrintList(struct List *list);
// Ausgeben aller Elemente der Liste
```

Die Funktionen sind in ihrem Interface so ausgelegt, daß sie mit einer beliebigen Anzahl von Listen arbeiten können (notwendig für abstrakte Datentypen).

Schreiben Sie ein Testprogramm, das eine komfortable Auswahl der Listenfunktionen zuläßt.

### Beispiel 2: Überladene Funktionen (12 Punkte)

Schreiben Sie eine Funktion `spur()`, die die Spur einer quadratischen Matrix – also die Summe der Elemente der Hauptdiagonale – berechnet. `spur` soll so überladen werden, daß Matrizen mit `int`- und mit `double`-Komponenten als Argumente möglich sind. Übergeben Sie die Matrizen jeweils mittels Zeiger auf Konstante.

Die Matrizen und die Ausgabe der Berechnung sollen in ansprechender Form dargestellt werden.

**Standards:**

Es gelten die üblichen Standards.

**Abgabe:** bis 12.11. 23:59 Uhr elektronisch (Verzeichnis uebung2 einrichten!) und am 13.11. zu Beginn der LVA (Listing der Quelldateien).