

## Hausübung 4 (Teil B der Abgabe), 30min Bearbeitung, MI1300

1. Laden Sie sich die Quelldatei `task.c` aus TUWEL herunter.
2. Passen Sie die Quelldatei nach den Anforderungen der Aufgabenstellung an.
3. Abgabe: Laden Sie die abgeänderte Quelldatei wieder als `task.c` in TUWEL hoch.
  - Sie dürfen keine andere Quelldatei öffnen.
  - Sie dürfen keinen Quellcode aus anderen Quellen/Dateien kopieren.
  - Sie dürfen *googeln*; aber auch im Browser dürfen Sie keine anderen Quelldateien öffnen oder betrachten.

---

Gegeben ist die Implementierung `struct HashTable` (im folgenden *Tabelle*) analog zur Hausübung (`HashTable.h/HashTable.c`). Eine Beschreibung der Struktur samt zugehöriger verfügbarer Funktionen steht in `HashTable.pdf` zur Verfügung.

Erweitern Sie das Programm in `task.c` mit der Implementierung der unten spezifizierten Funktion.

### Empfohlener Kompilierbefehl/Ausführen:

```
gcc -std=c11 -fsanitize=address -Wall -pedantic -g HashTable.c Book.c task.c && ./a.out
```

### Funktion `hashtable_erase`: Eintrag aus der Tabelle entfernen (15 Punkte)

```
bool hashtable_erase(struct HashTable *self, const T *value);
```

Funktionsbeschreibung: Sucht nach dem übergebenen Wert in der Tabelle. Wird ein Eintrag mit identischem Schlüssel gefunden wird der Eintrag entfernt. Zum Berechnen des Hashwertes wird die Funktion `self->hash` verwendet. Um zu überprüfen ob ein Eintrag dem gesuchten Wert/Schlüssel entspricht, wird die Funktion `self->compare` verwendet.

Parameter:

- `self`: Zeiger auf eine dynamisch allozierte Tabelle.
- `value`: Zeiger auf den einzufügenden Wert (Wertepaar).

Rückgabewert: `true` falls der Schlüssel gefunden und der zugehörige Eintrag entfernt wurde, und `false` falls der Schlüssel nicht vorhanden war.

### Tests in der `main`-Funktion:

Die `main`-Funktion ist als Test mit *asserts* (Annahmen) gestalten und soll Sie bei der Implementierung unterstützen. Sie müssen in der `main`-Funktion keine Anpassungen vornehmen, können aber jederzeit nach Bedarf den Quellcode zu Testzwecken anpassen/ändern/erweitern.