

☐ **Gr. 1, DI (FH) G. Horn, MSc**☐ **Gr. 2, J.-P. Haslinger, MSc****Name** _____ **Aufwand in h** _____**Punkte** _____ **Kurzzeichen Tutor / Übungsleiter** _____ / _____

1. Verwaltung von – vielen – Arbeitszeitaufzeichnungen (12 + 12 Punkte)

Gesucht ist erneut ein System zur Verwaltung von Arbeitszeitaufzeichnungen, das wiederum die bereits in der letzten beschriebene Funktionalität bietet. Diesmal soll allerdings die Anzahl der speicherbaren Einträge nicht mehr begrenzt sein, wodurch sich der Einsatz einer dynamischen Datenstruktur wie z. B. einer Liste zur Speicherung der Ergebnisse ja förmlich aufdrängt... ☺

a) Implementieren Sie ein neues Modul für Ihre Arbeitszeitaufzeichnungen oder bauen Sie Ihr bestehendes Modul aus der letzten Übung so um, dass nun für die interne Datenspeicherung eine einfach verkettete Liste nach folgender Definition zum Einsatz kommt:

TYPE

```
TimeRegistrationEntry = ^TimeRegistrationEntryRec;  
TimeRegistrationEntryRec = RECORD  
    ... (* actual data of time registration entry *)  
    next: TimeRegistrationEntry;  
END;  
TimeRegistrationEntryList = TimeRegistrationEntry;
```

Die Schnittstelle des neuen Moduls soll dabei im Vergleich zur letzten Übung möglichst unverändert bleiben, sodass die beiden Lösungen aus Sicht des Verwenders theoretisch jederzeit und (fast) ohne Aufwand ausgetauscht werden können. Dokumentieren Sie in Ihrer Lösungsidee aber auch, wo sich Ihrer Meinung nach die alte Schnittstelle vielleicht doch ein wenig ändern sollte.

Testen Sie dieses neue Modul ausführlich, auch mit den Testfällen aus der letzten Übung. Zeigen Sie aber nun auch die Vorteile, die sich durch den Einsatz einer dynamischen Datenstruktur ergeben, und beschreiben Sie die Unterschiede der beiden Lösungen.

b) Implementieren Sie zusätzlich noch eine Funktion **FUNCTION** BusiestPerson: STRING, die den Namen der Person mit den meisten Zeiterfassungseinträgen zurückliefert.

Bauen Sie dazu direkt während dem Aufruf der Funktion eine neue, temporäre Hilfsliste auf, in der Sie pro Person einen Knoten mit der Anzahl ihrer Zeiterfassungseinträge speichern. Suchen Sie danach die Person mit der höchsten Anzahl aus dieser Liste. Geben Sie schließlich die Liste wieder vollständig frei, bevor die Funktion verlassen wird.

Testen und kritisieren Sie die Performanz Ihrer Lösung: Was daran gefällt Ihnen bzw. gefällt Ihnen nicht? Lässt sich diese Aufgabe effizienter lösen?