

## **Aufgabe zur APL in Programmierung 1 im WS 2017/2018**

Die APL (Alternative Prüfungsleistung) besteht aus einem eigenständig zu bearbeitenden Programmierprojekt.

Es ist ein C-Programm zu erstellen, das Buchungen von Ferienhäusern verwaltet. Es sollen drei verschiedene Hauskategorien (A, B und C) angeboten werden. In jeder Kategorie sind 10 Ferienhäuser im Angebot. Die einzelnen Häuser werden durch ihre Kategorie und eine Nummer von 1 bis 10 identifiziert. In den Anfragen und Buchungen sind kalendarische Datumsangaben (Tag, Monat, Jahr) für Beginn und Ende des Buchungszeitraums anzugeben. Eine Buchung belegt das Haus ab Mittag 12 Uhr des ersten Buchungstags bis Mittag 12 Uhr des letzten Tags im Buchungszeitraums. Die Datumsangaben sind vor der Verarbeitung durch das Buchungssystem durch das Programm auf ihre Gültigkeit zu prüfen.

Das Buchungssystem soll die folgenden Operationen für Kunden unterstützen:

- Anfrage, ob Häuser in einer gewünschten Kategorie für einen Zeitraum frei sind. Es soll die Anzahl der freien Häuser und eine Auflistung der freien Häuser ausgegeben werden.
- Anfrage, ob ein gewünschtes Haus (Kategorie und Nummer) für einen Zeitraum frei ist. Es soll frei oder belegt ausgegeben werden.
- Buchung eines gewünschten Hauses (Kategorie und Nummer) für einen Zeitraum. Bei der Buchung ist die Kundennummer (7stellige Zahl) des Buchenden anzugeben und zu speichern. Es sollen nur Buchungen ausgeführt werden, wenn das Haus im angegebenen Zeitraum frei ist und die Buchungsdauer mindestens 3 Tage beträgt. Es soll die Annahme, bzw. die Ablehnung der Buchung signalisiert werden. Für jede erfolgreiche Buchung ist durch das Programm eine Buchungsnummer zu vergeben und auszugeben.
- Stornierung einer Buchung mit Angabe der Kundennummer und der Buchungsnummer. Bei Stornierung muss die Buchung gesucht werden und nur dann tatsächlich eine Stornierung vorzunehmen, wenn Kundennummer und Buchungsnummer mit der Buchung übereinstimmen.

In der abgegebenen Endversion Ihres Programms sind die Daten programmintern durch Datenstrukturen in dynamischen Speicherbereichen zu verwalten. Die Daten sollen bei Programmende auf eine Datei geschrieben werden, die bei einem wiederholten Programmstart wieder eingelesen wird. Damit bleiben die Buchungsdaten über mehrere Programmausführungen erhalten.

Für den Betreiber des Buchungssystems sollen darüber hinaus ausgewählte Funktionen angeboten werden, die von Studierenden individuell zu bearbeiten sind.

Dazu wird jede/r Studierende einer Gruppe (A bis D) zugeordnet (siehe unten) und hat die Funktion/en für seine Gruppe zu programmieren.

A: Studierende mit gerader Matrikelnummer und gerader s-Nummer.

B: Studierende mit gerader Matrikelnummer und ungerader s-Nummer.

C: Studierende mit ungerader Matrikelnummer und gerader s-Nummer.

D: Studierende mit ungerader Matrikelnummer und ungerader s-Nummer.

## Aufgaben für Gruppen

### Gruppe A:

Es sind Operationen für das Anzeigen einer Rechnung zu einer Buchung und das Vermerken der Bezahlung für eine Buchung bereitzustellen. Die Preise je Übernachtung sollen als konstante Werte für die verschiedenen Hauskategorien im Quellcode vorgegeben werden, z.B. 129 Euro für A, 99 Euro für B und 89 Euro für C. Das für Gruppe C beschriebene Preismodell nach Saison muss nicht berücksichtigt werden. Darüber hinaus ist die Entfernung aller bezahlter Buchungen aus dem Buchungssystem zu ermöglichen, deren Ende des Buchungszeitraums zeitlich vor einem einzugebenden Datum liegt.

### Gruppe B:

Es soll der Belegungsanteil in Prozent für einen angegebenen Zeitraum (von Datum, bis Datum) nach Hauskategorien gruppiert ermittelt werden. Zur Wertung eines im Zeitraum liegenden Tag ist die Belegung der Übernachtung zum Folgetag ausschlaggebend.

### Gruppe C:

Zu jeder Anfrage und Buchung ist ein Preis zu berechnen, auszugeben und im Falle einer Buchung zu speichern. Der Preis wird nach einem vorgegebenen Preismodell berechnet, das auch abhängig von der Saison ist, in der die Buchung liegt.

Buchungen kosten je Nacht die folgenden Beträge:

	Wintersaison 1.11.-14.3.	Frühjahrssaison 15.3.-30.4.	Sommersaison 1.5.-30.9.	Herbstsaison 1.10.-31.10.
Kategorie A	99,- Euro	119,- Euro	149,- Euro	129,- Euro
Kategorie B	79,- Euro	89,- Euro	109,- Euro	99,- Euro
Kategorie C	69,- Euro	79,- Euro	99,- Euro	89,- Euro

Für die Wahl der Saison ist immer der einer Übernachtung vorhergehende Tag ausschlaggebend, d.h. zum Beispiel kostet die Übernachtung vom 30.9. zum 1.10. im Kategorie B noch 109,- Euro. Ab einer Buchung von mehr als 8 aufeinanderfolgenden Nächten wird ab der 9. Nacht ein Nachlass von 20 Euro pro Nacht gewährt.

### Gruppe D:

Es sind die im System gespeicherten Buchungen nach Hauskategorie (A, B, und C) und nach vier Buchungsdauerklassen (3 bis 4 Tage, 5<sup>1</sup> bis 7 Tage, 8 bis 9 Tage, 10 oder mehr Tage) durch das Programm zu zählen. Die Anzahl der Buchungen für jede Kategorie und Buchungsdauerkategorie, sowie der prozentuale Anteil gegenüber allen Buchungen ist auszugeben.

<sup>1</sup> Geändert von 6 auf 5 wegen eines Tippfehlers (5.12.2017, P. Sobe).

## **Bewertung**

Die APL wird mit maximal 30 Punkten bewertet. Die erreichte Punktzahl wird auf eine Note abgebildet.

Sie müssen die Aufgabe selbständig bearbeiten. Lösungen die offensichtlich von anderen Studierenden kopiert wurden, können automatisiert erkannt werden und können je nach Umfang des Kopierens zu Punktabzug führen.

In die Bewertung fließt zu einem Teil die Erklärung des Programmentwurfs und die Vorführung der ersten Programmversion ein. Der andere Teil der Bewertung wird ausgehend vom abgegebenen Programm in seiner Endfassung vorgenommen.

Das Programm wird nach funktionalen Kriterien und nach nicht-funktionalen Kriterien bewertet. Die funktionalen Kriterien umfassen das Vorhandensein der vollständigen Funktionalität laut Aufgabenstellung, sowie die Fehlerfreiheit bei der Ausführung.

Nicht-funktionale Kriterien umfassen beispielsweise eine sinnvolle Gliederung des Programms und der Daten, die Abwesenheit von Code-Dopplungen, eine fehlerfreie Programmierung, sowie das Vorhandensein geeigneter Datenstrukturen im dynamischen Speicher.