



Project: Car-Activation

Kjell Wagner & Cedric Wirbs

Car-Activation V1.0

Lese Pools ein...

Bitte wähle eine Option aus:

- 1 Huffmann-Baum ausgeben
- 2 Pool Daten neu einlesen
- 3 Pooldaten ausgeben
- 4 Alle Buchungen ausgeben
- 5 Neue Buchung einlesen
- 6 Programm verlassen

Deine Eingabe: 5

Bitte Buchungsname angeben:

booking1.txt

Buchung angenommen!

Designspezifikationen für das Projekt “Car-Activation”

1. Was, wie und warum

In einer zunehmend überfüllten Stadtlandschaft werden Parkplätze immer knapper. Öffentliche Verkehrsmittel oder Fahrräder sind jedoch nicht immer geeignet, um Möbel zu transportieren oder größere Gruppen zu befördern. Car-Sharing hat sich als kostengünstige Möglichkeit etabliert, um von A nach B zu gelangen.

Als Anbieter einer Car-Sharing-Lösung stellt die Instandhaltung der Fahrzeuge, rechtliche Angelegenheiten und der Kundenservice eine große Herausforderung dar.

Unser Programm "Car-Activation" revolutioniert den Buchungsprozess. Durch die bequeme Eingabe von Buchungsanfragen über die Kommandozeile vergleicht die Software die Anfrage mit aktuellen Kundendaten, Fahrzeuginformationen und Ausleihdaten, um Kollisionen zu prüfen. Dadurch vereinfacht unsere Lösung den Ablauf erheblich!

2. Hauptziele

Was unser Produkt erreichen möchte:

- Einfache Verwaltung von Buchungsanfragen: Unsere Lösung ermöglicht eine benutzerfreundliche Verwaltung von Buchungsanfragen, wodurch der Arbeitsaufwand erheblich reduziert wird.
- Visuelle Bestätigung oder Ablehnung bei Fehlern: Durch visuelle Rückmeldungen erhalten Sie sofortige Bestätigung oder Ablehnung von Buchungsanfragen bei Fehlern, was zu einer effizienteren Bearbeitung führt.
- Aufzeigen der Fehler in Buchungsanfragen: Unsere Lösung zeigt klar und deutlich die spezifischen Fehler in den Buchungsanfragen auf, sodass sie schnell und präzise behoben werden können.
- Reduzierter Arbeitsaufwand: Durch die Automatisierung des Prüfprozesses entlasten wir Ihre Mitarbeiter von manuellen Überprüfungen und ermöglichen ihnen, sich auf andere Aufgaben wie den Kundensupport zu konzentrieren.
- Einfache Skalierbarkeit: Unsere Lösung ist einfach skalierbar und kann an die wachsenden Anforderungen Ihres Car-Sharing-Unternehmens angepasst werden, ohne die Effizienz zu beeinträchtigen.
- Reduzierter Speicherplatz: Durch die Verwendung aktueller Pooldaten zur Laufzeit wird der Speicherplatzbedarf optimiert und unnötige Redundanzen vermieden.

Durch die Implementierung unserer Applikation wird der Arbeitsaufwand erheblich reduziert. Die manuelle Prüfung von Anfragen und die Verwaltung der Autos werden überflüssig. Die Buchungsverifikation erfolgt mit nur einer Eingabe in die Kommandozeile mithilfe unserer Lösung. Die automatische Verhinderung von Terminkollisionen

ermöglicht Ihren Mitarbeitern, sich auf andere Aufgaben zu konzentrieren, insbesondere den Kundensupport. Darüber hinaus führt die verbesserte Fehlererkennung zu einer höheren Kundenzufriedenheit, da potenzielle Probleme bei Buchungen schnell identifiziert und kommuniziert werden können.

3. Zeitbudget

Für das Projekt "Car-Activation" haben wir als Entwickler ein Zeitbudget von vier Wochen festgelegt, um eine erste Präsentation für Sie als Kunden vorzubereiten. In diesen vier Wochen werden wir mehrere Sprints abschließen, um die erforderlichen Funktionen und Merkmale zu implementieren und eine solide Grundlage für die Präsentation zu schaffen.

Nach diesen vier Wochen werden wir bereits folgende Punkte auf unserer Roadmap erreicht haben:

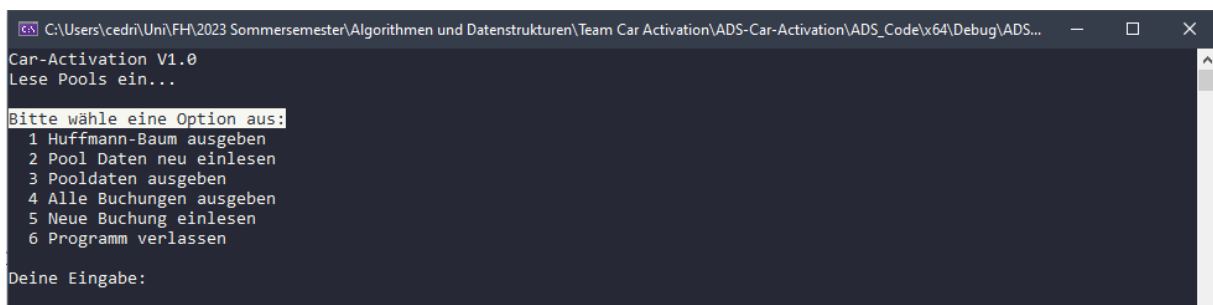
- Implementierung von mehreren Anfragen in einer Datei (Batch Verarbeitung): Diese Funktion ermöglicht die Verarbeitung von mehreren Buchungsanfragen in einer einzigen Datei, um den Prozess effizienter zu gestalten.
- Reduzierung der benötigten Daten durch eine effiziente Kodierung: Wir werden eine effiziente Kodierung implementieren, um den Speicherplatzbedarf zu reduzieren und die Systemleistung zu optimieren.

Darüber hinaus werden wir nach der ersten Präsentation auf Ihre spezifischen Anforderungen und Wünsche eingehen. Wir sind bestrebt, Ihre Feedbacks in die weitere Entwicklung und Verbesserung der Software einfließen zu lassen.

4. Funktionalität

Buchungseingabe und Verwaltung

Die Software wird über eine Kommandozeile gesteuert. Zum Start werden zunächst die Pooldaten eingelesen. Anschließend wird der Benutzer nach seiner Aktion gefragt. Die Eingabe erfolgt mit einer Ziffer von 1 bis 6 mit anschließendem Bestätigen durch die Eingabe-Taste. Eine falsche Eingabe führt erneut ins Auswahlm Menü.



```
C:\Users\cedri\Uni\FH\2023 Sommersemester\Algorithmen und Datenstrukturen\Team Car Activation\ADS-Car-Activation\ADS_Code\Debug\ADS...
Car-Activation V1.0
Lese Pools ein...

Bitte wähle eine Option aus:
1 Huffmann-Baum ausgeben
2 Pool Daten neu einlesen
3 Pooldaten ausgeben
4 Alle Buchungen ausgeben
5 Neue Buchung einlesen
6 Programm verlassen

Deine Eingabe:
```

Zur Verfügung steht:

1. Huffman-Baum ausgeben
2. Pool Daten neu einlesen

3. Pooldaten ausgeben
4. Alle Buchungen ausgeben
5. Neue Buchung einlesen
6. Programm verlassen

1. Huffman-Baum ausgeben

Der von uns verwendete Huffman-Algorithmus ist ein effizientes Verfahren zur verlustfreien Datenkompression, das entwickelt wurde, um Informationen auf elegante Weise zu speichern und zu übertragen. Bei der Datenkompression geht es darum, die Menge an Speicherplatz oder Übertragungsbandbreite zu reduzieren, ohne dabei Informationen zu verlieren. Der Huffman-Algorithmus erfüllt dieses Ziel, indem er die am häufigsten vorkommenden Symbole in den Daten identifiziert und ihnen kürzere Codewörter zuweist, während seltener vorkommende Symbole längere Codewörter erhalten.

Der Algorithmus funktioniert, indem er zunächst eine Häufigkeitstabelle erstellt, die angibt, wie oft jedes Symbol in den Daten vorkommt. Anschließend werden diese Symbole basierend auf ihrer Häufigkeit sortiert. Der Huffman-Algorithmus nutzt eine Baumstruktur, um die Symbole zu codieren. In jedem Schritt werden die beiden Symbole mit der niedrigsten Häufigkeit ausgewählt und zu einem neuen Knoten zusammengefasst. Dieser neue Knoten wird dann in den Sortierungsprozess eingefügt, und dieser Vorgang wird wiederholt, bis alle Symbole im Baum enthalten sind.

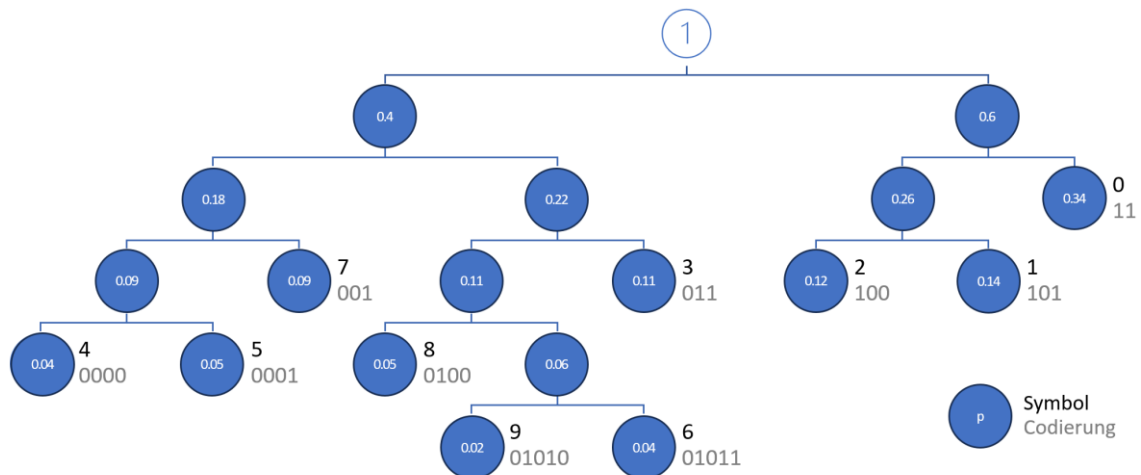
Dieses Verfahren wird bei jedem Programmstart, sowie beim Aufrufen der Funktion '1' -> Huffman-Baum ausgeben, ausgeführt:

```
Huffmann-Baum wird ausgegeben.  
Ausgabe der Wahrscheinlichkeiten:  
Symbol 0 0.34  
Symbol 1 0.14  
Symbol 2 0.12  
Symbol 3 0.11  
Symbol 4 0.04  
Symbol 5 0.05  
Symbol 6 0.04  
Symbol 7 0.09  
Symbol 8 0.05  
Symbol 9 0.02
```

Zu Beginn zählt die Funktion die Vorkommenden Symbole in den Autocodierungen. Zur Grundlage dient hierbei der zur Verfügung gestellte Datenbank „car.pool“. Die Funktion gibt anschließend die prozentuale Wahrscheinlichkeit jedes Symbols aus. Anhand dieser Wahrscheinlichkeiten wird im nächsten Schritt der Huffman-Baum erzeugt:

```
binäre 0 im Baum nach links | binäre 1 im Baum nach rechts  
  
Symbol  Codierung  
4        0000  
5        0001  
7        001  
8        0100  
9        01010  
6        01011  
3        011  
2        100  
1        101  
0        11
```

Etwas ersichtlicher wird der Baum, wenn dieser anhand der oben gegebenen Codierung zusammen mit den Wahrscheinlichkeiten notiert wird. Es entsteht ein aus der Informatik vertrauter Baum:



Die Codierung erfolgt, indem man den Huffman-Baum von der Wurzel bis zu den Blättern durchläuft. Jedes Mal, wenn man sich nach links bewegt, wird ein '0' hinzugefügt, und jedes Mal, wenn man sich nach rechts bewegt, wird ein '1' hinzugefügt. Jedes Symbol hat nun eine eindeutige Codierung, die auf dem Pfad vom Baumwurzel bis zum Blatt basiert.

Durch die Verwendung des Huffman-Algorithmus können die Daten effizient komprimiert werden, da häufig vorkommende Symbole weniger Bits zur Darstellung benötigen als seltener vorkommende Symbole. Dadurch wird der Speicherplatzbedarf reduziert und die Übertragungszeit verkürzt, da weniger Daten übertragen werden müssen. Beim Dekomprimieren können die Codewörter mithilfe des Huffman-Baums wieder in die ursprünglichen Daten umgewandelt werden, wobei keine Informationen verloren gehen.

2. Pool Daten neu einlesen

Ein Car-Sharing Dienst wächst schnell. Neue Kunden, neue Standorte, neue Autos. Unser Dienst kann ohne Down-Time betrieben werden, da zur Laufzeit alle Pools neu eingelesen werden können.

3. Pool Daten ausgeben

Ist sich der Anwender nicht sicher, welchen Stand die Pool Daten haben oder ob ein Kunde existiert, so kann er sich mittels '3' -> Enter den aktuellen Pool ausgeben.

Deine Eingabe: 3

Kundenname	Kundennummer
Linus Torvald	1
Konrad Zuse	2
Bill Gates	3
Steve Wozniak	4
Steve Jobs	5
Ada Lovelace	6
Mark Zuckerberg	7
Grace Hopper	8
Larry Page	9
James Gosling	10
Dirk Nowitzki	11
Christiano Ronaldo	12
Mike Tyson	13
Wayne Gretzky	14
Tom Brady	15

StationsID	Station
1	H-KA
2	PH
3	KIT
4	Marktplatz
5	Europaplatz
6	Durlacher Tor
7	Bahnhof
8	Duale Hochschule
9	Schlossplatz
10	KIT Nord
11	KIT West
12	Musikhochschule
13	H-KA Technologiepark
14	Kunstakademie

Autoname	Kategorie	Kodierung	Preis/h	Preis/km
CruiserOne	Cat1	1012607	0.50	1.50
CruiserOne	Cat1	1024353	0.50	1.50
CruiserOne	Cat1	1035001	0.50	1.50
CruiserTwo	Cat2	2018877	1.25	2.50
CruiserTwo	Cat2	2029504	1.25	2.50
CruiserThree	Cat3	3011873	2.50	5.00
CruiserThree	Cat3	3022607	2.50	5.00

4. Alle Buchungen ausgeben

Eine Ausgabe aller bisher erfassten Buchungen, z.B. wenn sich ein Kunde nicht mehr sicher ist, erfolgt über die Funktion "4" - "Alle Buchungen ausgeben". Durch Auswahl dieser Option erhalten sie eine vollständige Liste aller Buchungen, die bisher im System erfasst wurden. Die Ausgabe umfasst alle relevanten Parameter jeder Buchung, um eine umfassende Übersicht über die getätigten Buchungen zu geben.

Bitte wähle eine Option aus:

- 1 Huffmann-Baum ausgeben
- 2 Pool Daten neu einlesen
- 3 Pooldaten ausgeben
- 4 Alle Buchungen ausgeben
- 5 Neue Buchung einlesen
- 6 Programm verlassen

Deine Eingabe: 4

Alle Buchungen werden ausgegeben.

Kundenname	KundenID	Startzeitpunkt	Endzeitpunkt	Startpunkt	Endpunkt	Kategorie	Autocodierung
Mark Zuckerberg	0000007	20.06.23 18:40	20.06.23 20:40	H-KA	Bahnhof	Cat1	1012607
Torvald	0000001	25.06.23 10:40	30.06.23 10:40	PH	KIT Nord	Cat3	3022607
Wozniak	0000003	01.07.23 18:40	02.07.23 20:40	H-KA Technologiepark	Musikhochschule	Cat2	2018877

5. Neue Buchung einlesen

Die Buchungen erfolgen über ein vom Produktbesitzer festgelegtes Format in einer Textdatei. Um sicherzustellen, dass das Format eingehalten wird, führt unsere Anwendung eine Überprüfung durch. Sollte das Format nicht korrekt sein, wird eine entsprechende Fehlermeldung ausgegeben. Weitere Informationen dazu werden in einem späteren Absatz behandelt.

Damit die Buchungen richtig erfasst werden können, muss die Buchungsdatei sich im selben Verzeichnis wie der Programmcode befinden. Die Eingabe erfolgt, indem der Dateiname einschließlich der Dateiendung eingegeben wird, gefolgt von der Bestätigung durch die Eingabetaste (Enter).

Eine erfolgreiche Ausgabe könnte wie folgt aussehen:

```
Bitte wähle eine Option aus:
1 Huffmann-Baum ausgeben
2 Pool Daten neu einlesen
3 Pooldaten ausgeben
4 Alle Buchungen ausgeben
5 Neue Buchung einlesen
6 Programm verlassen

Deine Eingabe: 5

Bitte Buchungsname angeben:
booking2.txt
Buchung angenommen!
```

Der eingegebene Zeitpunkt der Buchung wird intern auf einen Unix-Zeitcode umgerechnet, um eine effiziente Überprüfung von Überschneidungen zu ermöglichen. In dem untenstehenden Screenshot unserer Anwendung wird zunächst die Ausgabe aller Buchungen angezeigt. Die neu eingegebene Buchung "booking1.txt" versucht, dasselbe Auto für einen überlappenden Zeitraum zu buchen. Allerdings wird die Buchung abgelehnt, da unsere Applikation eine Terminüberschneidung festgestellt hat. Die rot markierte Ausgabe zeigt deutlich an, dass etwas nicht stimmt, und gibt zusätzlich die Information aus, dass das Auto bereits gebucht ist.

```
Bitte wähle eine Option aus:
1 Huffmann-Baum ausgeben
2 Pool Daten neu einlesen
3 Pooldaten ausgeben
4 Alle Buchungen ausgeben
5 Neue Buchung einlesen
6 Programm verlassen

Deine Eingabe: 4

Alle Buchungen werden ausgegeben.
Kundenname      KundenID Startzeitpunkt Endzeitpunkt Startpunkt Endpunkt Kategorie Autocodierung
Mark Zuckerberg  0000007  20.06.23 18:40  20.06.23 20:40 H-KA      Bahnhof      Cat1      1012607

Bitte wähle eine Option aus:
1 Huffmann-Baum ausgeben
2 Pool Daten neu einlesen
3 Pooldaten ausgeben
4 Alle Buchungen ausgeben
5 Neue Buchung einlesen
6 Programm verlassen

Deine Eingabe: 5

Bitte Buchungsname angeben:
booking1.txt
Buchung muss abgelehnt werden! Grund: Leider ist das Auto zu dem angegebenen Zeitpunkt schon ausgebucht!
```

Durch die farbliche Hervorhebung und klare Fehlermeldung ermöglicht unsere Applikation eine schnelle und einfache Identifizierung von Buchungsproblemen. Dadurch wird vermieden, dass es zu Doppelbuchungen kommt und der Buchungsprozess reibungslos abläuft. Unsere Lösung bietet eine effektive Methode zur Vermeidung von Terminüberschneidungen und gewährleistet, dass jeder Kunde seine Buchungen ohne Schwierigkeiten vornehmen kann.

5. Fehler Erkennung

Unser System überwacht sorgfältig alle Vorgänge auf mögliche Fehler und stellt sicher, dass die eingegebenen Daten korrekt sind. Es werden verschiedene Fehlerarten erkannt und entsprechende Fehlermeldungen ausgegeben, um den Benutzer auf Probleme hinzuweisen. Hier sind die möglichen Fehler, die erkannt und ausgegeben werden können:

1. NoError: Kein Fehler ist aufgetreten. Die Daten wurden erfolgreich verarbeitet. Dies wird nicht ausgegeben, sondern resultiert in einer erfolgreichen Buchungsbestätigung.
2. NotEnoughInputs: Es wurden nicht genügend Eingaben gemacht oder die Datei enthält nicht genügend Informationen. Der Benutzer wird darauf hingewiesen, dass weitere Informationen erforderlich sind.
3. InvalidFormat: Das Format der eingegebenen Datei entspricht nicht den erwarteten Vorgaben. Es wird darauf hingewiesen, dass das Format überprüft und angepasst werden muss.
4. StringMismatch: Ein eingegebener Wert stimmt nicht mit dem erwarteten Textformat überein.
5. IntegerMismatch: Ein eingegebener Wert stimmt nicht mit dem erwarteten Zahlenformat überein.
6. FileNotFound: Die angegebene Datei wurde nicht im Verzeichnis gefunden.

7. NoCustomerFound: Es konnte kein Kunde mit der angegebenen Kundennummer oder dem Kundennamen gefunden werden.
8. NoCarFound: Das angegebene Auto wurde nicht gefunden oder die Kategorie bzw. Codierung des Autos stimmt nicht mit dessen Name überein.
9. NoStationFound: Die angegebene Station wurde nicht gefunden.
10. AlreadyBooked: Die Buchungsanfrage überschneidet sich mit einer bereits bestehenden Buchung für dasselbe Auto.

Durch die Erkennung und gezielte Ausgabe dieser Fehlermeldungen wird die Datenintegrität gewährleistet und der Benutzer erhält eine klare Rückmeldung bei fehlerhaften Eingaben.

6. Design

Im Hinblick auf das Design haben wir verschiedene Maßnahmen ergriffen, um eine verbesserte optische Erkennbarkeit und Benutzerfreundlichkeit zu gewährleisten:

- Prägnante Farben wurden gewählt, um wichtige Informationen hervorzuheben und eine bessere visuelle Trennung zu ermöglichen.
- Wir haben uns für die Darstellung der Ausgaben in Tabellenform entschieden. Diese Formatierung trägt dazu bei, die Informationen übersichtlich und strukturiert darzustellen.

Durch diese Designanpassungen streben wir eine optimale Benutzererfahrung an, indem wir die Lesbarkeit, Übersichtlichkeit und visuelle Ästhetik unserer Software verbessern.

7. Zusammenfassung

Das Projekt Car-Activation zielt darauf ab, eine effiziente und benutzerfreundliche Car-Sharing-Lösung anzubieten. Durch die Entwicklung einer speziellen Software, die den Buchungsprozess vereinfacht und automatisiert, sollen verschiedene Ziele erreicht werden. Dazu gehört die einfache Verwaltung von Buchungsanfragen durch den Vergleich mit aktuellen Kundendaten, Autodaten und Ausleihstandorten. Fehlerhafte Anfragen werden visuell gekennzeichnet und die Ausgabe optimiert, um eine verbesserte Lesbarkeit und Übersichtlichkeit zu gewährleisten. Das System überwacht auch die Einhaltung des vorgegebenen Dateiformats und erkennt verschiedene Fehler wie ungültige Eingaben, fehlende Daten oder bereits gebuchte Autos. Durch die Implementierung einer effizienten Kodierung und Skalierbarkeit wird der Speicherplatz in späteren Versionen reduziert und die Anwendung kann mit wachsenden Datenmengen umgehen. Insgesamt konzentriert sich das Projekt darauf, die Buchungsverwaltung im Car-Sharing-Bereich zu optimieren und den Benutzern eine reibungslose und benutzerfreundliche Erfahrung zu bieten.