|  |  |
| --- | --- |
| **Assignment Case** |  |
| COMP6048  Data Structures |
| **Computer Science** | **E213-COMP6048-RR06-01** |
| ***Valid on*** *Even Semester Year 2021/2022* | **Revision 00** |

1. Seluruh mahasiswa tidak diperkenankan untuk:

*All students are not allowed to:*

* + - Berdiskusi dan/atau bekerja sama dengan mahasiswa lainnya

*Discuss and/or work together with other student participants*

* + - Melihat sebagian atau seluruh jawaban mahasiswa lain

*Seeing a part or the whole answer from another student*

* + - Membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya

*Open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references*

* + - Membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb)

*Open and copy answer from the internet (google, stackoverflow, etc)*

* + - Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal,

*Working with another theme which is not in accordance with the existing theme in the matter of the case,*

* + - Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + - Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika mahasiswa terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai mahasiswa** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the student is proved to the actions described in point 1 above, the score of the student which committed dishonest acts (cheating or being cheated) will be “Zero”*

1. Perhatikan jadwal pengumpulan jawaban, segala jenis pengumpulan jawaban di luar jadwal tidak dilayani.

*Pay attention to the submission schedule, all kinds of submission outside the schedule will not be accepted*

1. Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |
| --- | --- |
| **Tugas Mandiri**  *Assignment* | **UAP**  *Final Exam* |
| 40% | 60% |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| Dev-C++ 5.11 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri, dan uap untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment and final exam collection for this subject are described as follows:*

|  |  |
| --- | --- |
| **Tugas Mandiri**  *Assignment* | **UAP**  *Final Exam* |
| CPP | CPP |

## Soal

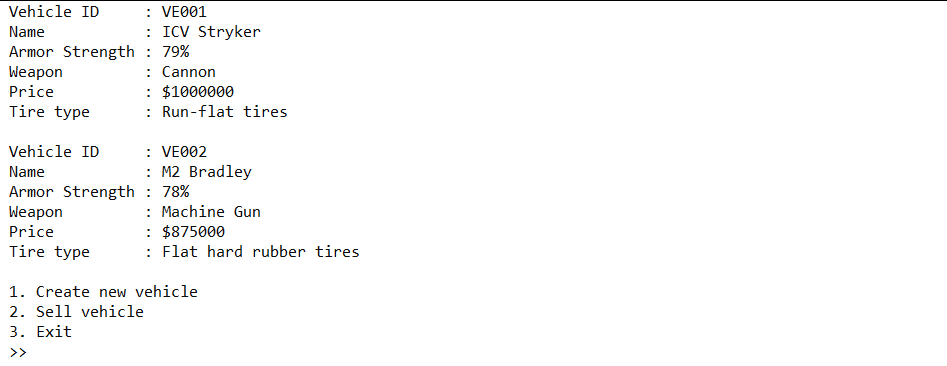
*Case*

**Military Car**

Cao-Cao has a vehicle factory that produce military car for war preparation, you as a programmer are asked to create a program that can maintain vehicle production list data using **C language** and **Chaining Hash Table** data structure. The program that will be created must follow the requirements below.

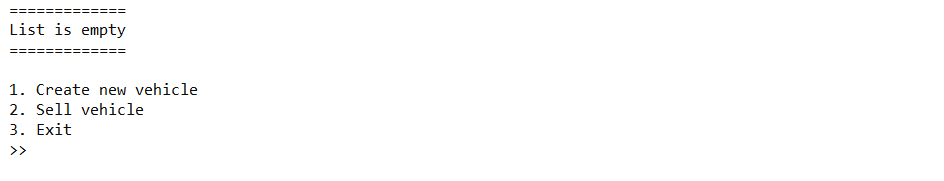
* The program will have **3 menus** and **show** **all produced vehicle**:

1. Create new vehicle
2. Sell vehicle
3. Exit



*Figure 1. Main Menu with Vehicles*

* If there are no vehicle yet or the list is empty, then the program **should show no vehicle message**.



*Figure 2. Main Menu with No Vehicle*

* If the user choose **menu 1** (**Create new vehicle**), then:

1. The program will **generate the Vehicle Id** for the inputted vehicle data. The Vehicle Id must follow the format below:

VEXXX

XXX : The last 3 digits of the last vehicle id added by one.

**Example**:

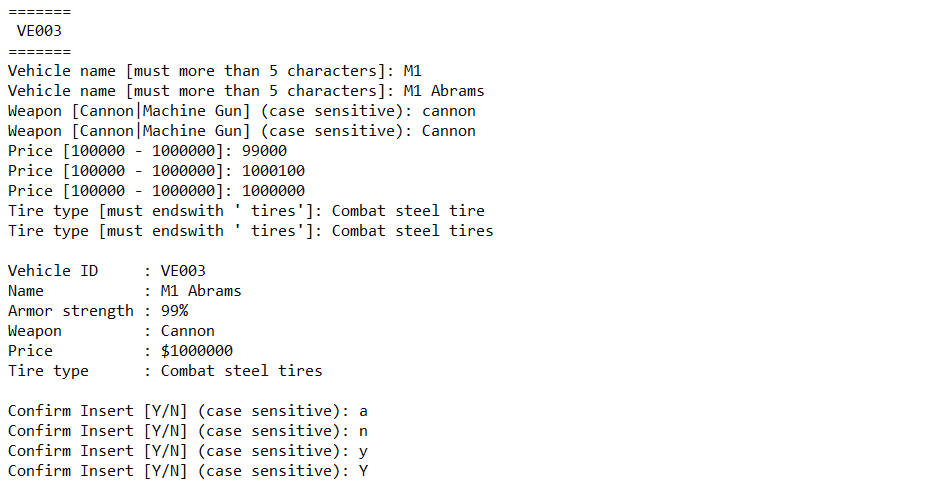
The last vehicle id is VE002

Then new generated vehicle id is VE003

*Figure 3. Vehicle Id Format*

1. The program will ask user to input the **name of vehicle** and **validate** that the **name** must **more than five** characters.
2. The program will **random** armour strength value for the vehicle, random **value is between 60 – 100** (**inclusive**).
3. The program will ask user to input the **vehicle weapon** and **validate** that the **weapon** must between “**Cannon**” or “**Machine Gun**” (**case sensitive**).
4. The program will ask user to input the **price** and **validate** that the **price** must be **between 100.000 and 1.000.000** (**inclusive**).
5. The program will ask user to input the **tire type** and **validate** that the **tire** must **ends with**

**“ tires”** without quotes.



*Figure 4. Input Validation*

1. The program will show vehicle summary and **ask user** **to** **confirm** inserting vehicle. User input must between **“Y”** or **“N” (case sensitive).** Below is the condition for each input.

* If input is “**Y**”, then the vehicle **will be inserted** and **show** **success message**.
* If input is “**N**”, then **cancel insert** action and **show** **cancelled message,** thenuser **will be directed back to the menu page**.



*Figure 5. Insert Success*



*Figure 6. Insert Cancelled*

1. The program will **store new vehicle data** to the **next item** of the **last item** of **chaining hash table** with **size** **100** using the following **hash function**.

Key = X % Y

Key : the hash table index that will store the data.

X : the last 3 digits of the vehicle id minus by one.

Y : size of the hash table (100).

**Example**:

Order Id : VE003

Size : 100

Key : (003 – 1) % 100

: 2

Then the vehicle data will store at index 2 of hash table

*Figure 7. Hash Function*

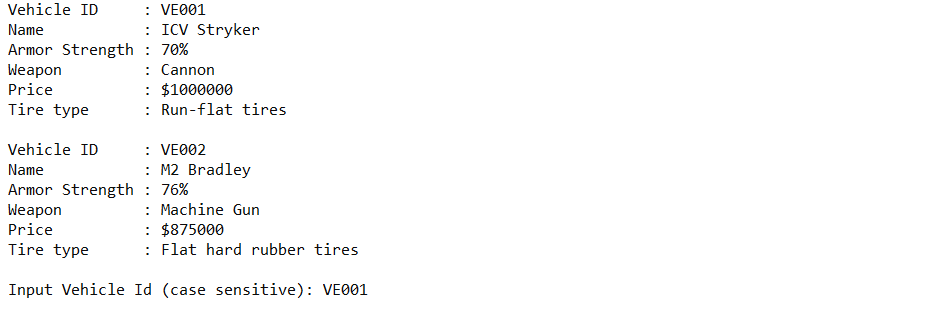
* If the user choose **menu 2** (**Sell vehicle**), then:

1. The program will **show all the vehicle list**. If there are no vehicle yet or the list is empty, then the program should **show no vehicle list message** and **redirect** **user back to menu page**.

****

*Figure 8. No Vehicle to Delete*

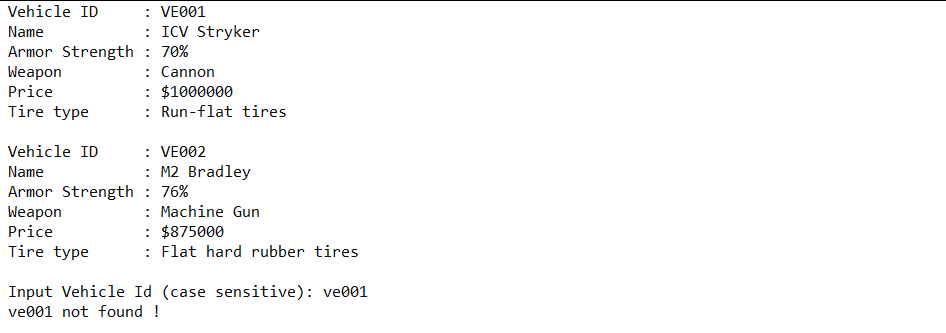
1. Otherwise, the program will ask user to input the **vehicle id to delete**.



*Figure 9. Input Vehicle Id*

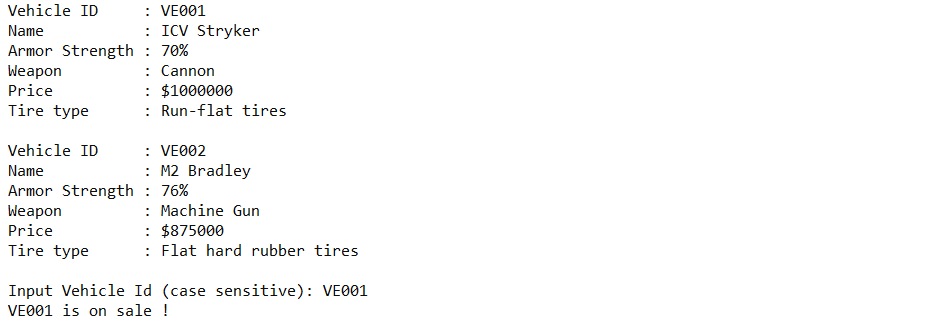
1. The program will **search** for the **specified vehicle id (case sensitive)**.

* If the **vehicle id** is **not found** the program will **show** the **Vehicle Id not found message** and **will be redirected back to menu page**.

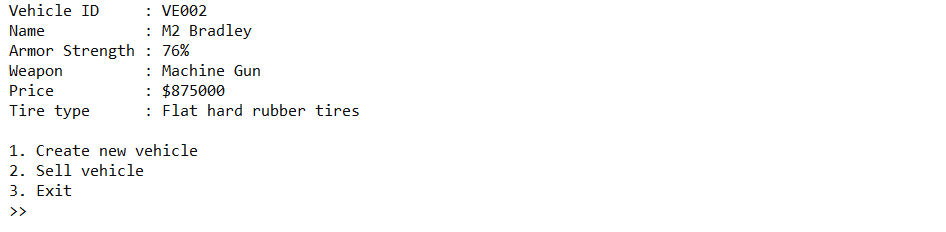


*Figure 10. Vehicle Not Found Message*

* If the **vehicle id exists**, then:
  + The program will **remove** the **vehicle data** from the **chaining hash table**
  + The program will **show “(Vehicle Id) is on sale!”** message.
  + The program **will be redirected to the menu page**.



*Figure 11. Vehicles data delete (VE001)*



*Figure 12. Vehicle list after item deleted (VE001)*

* If the user choose **menu 3** (**Exit**), then **exit from the program**.

**Please run the EXE file to get more detail about the application.**

Here are the rules that you must follow to create your project:

1. Use appropriate software for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
2. Use the techniques taught during practicum
3. Collect appropriate files for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
4. Include the other files that can support your project, such as:
   * All files in your project
   * Other files (image, audio, video, etc.) used in your project
   * \*.DOC file (documentation of your project) that contains the reference links of additional files (image, audio, video, etc.) used in your project