|  |  |
| --- | --- |
| **Assignment Case** |  |
| COMP6048  Data Structures |
| **Computer Science** | **E213-COMP6048-SL09-01** |
| ***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

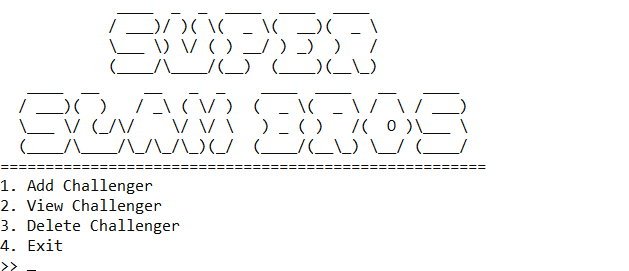
*Case*

**Super SLam Bros**

**Super SLam Bros** is a famous action fighting game. **Super SLam Bros** has a lot of challenger that want to join the battle. To increase the efficiency in **Super SLam Bros**, you as a programmer are asked to create a program that can maintain challenger data using **C language** and **Chaining Hash Table** data structure. The program that will be created must be following the below requirements.

* The program will have **4 menu items**:

1. Add Challenger
2. View Challanger
3. Delete Challanger
4. Exit



*Figure 1. Main Menu*

* If the user choose **menu 1** (**Add Challenger**), then:

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

CHXXX

XXX : the last 3 digits of the auto increment Challenger ID

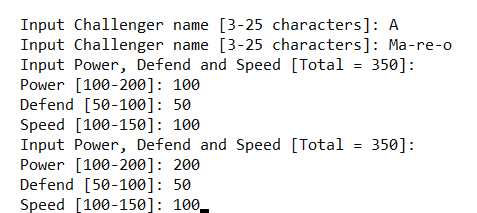
**Example**:

The last challenger id is CH003

Then the new generated challenger id is CH004

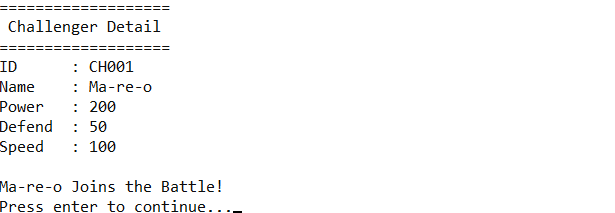
*Figure 2. Order ID Format*

1. The program will ask user to input the **name of challenger** and **validate** that the **name** must **between 3 and 25** characters (**inclusive**).
2. The program will ask user to input the **challenger power** and **validate** that the **power** must be **between 100 and 200** (**inclusive**).
3. The program will ask user to input the **challenger defend** and **validate** that the **defend** must be **between 50 and 100** (**inclusive**).
4. The program will ask user to input the **challenger speed** and **validate** that the **speed** must be **between 100 and 150** (**inclusive**).
5. The program will **validate** that **sum** of **challenger** **power**, **defend** and **speed** must be **exactly** **equals** to **350**



*Figure 3. Input Challenger Validation*

1. The program will show the **challenger** **detail, join the battle message, store the data**, and then user **will be directed back to the menu page**.



*Figure 4. Challenger Detail*

1. The program will **store new challenger data** to the **next item** of the **last item** of **chaining hash table** with **size** **10** using the following **hash function**.

Key = (X / Y) % Y

Key : the hash table index that will store the data

X : the last 3 digits of the challenger id

Y : size of the hash table (10)

**Example**:

Challenger Id : CH003

Size : 10

Key : (003 / 10) % 10

: 0

Then the challenger data will store at index 0 of hash table

*Figure 5. Hash Function*

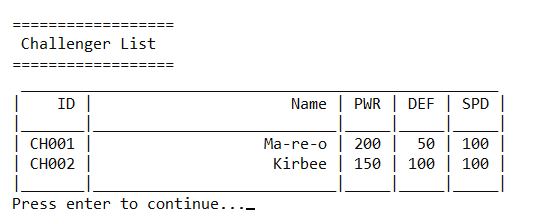
* If the user choose **menu 2** (**View Challenger**), then:

1. If there are no challenger yet or the list is empty, then the program **should show no data message** “**No Data Found!**” and **redirect** user back to menu page.



*Figure 6. No Data Found Message*

1. Otherwise, the program will **show all the challenger list**.



*Figure 7. Challenger List*

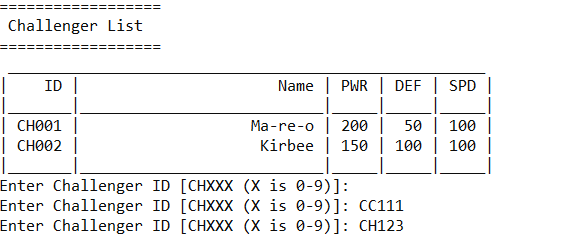
* If the user choose **menu 3** (**Delete Challenger**), then:

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



*Figure 8. No Data Found Message*

1. Otherwise, the program will ask user to input the **challenger id to delete**. **Validate** the **challenger id** format **must** **starts with** “**CH**” and **ends with 3 digits of number** (with format **CHXXX**, where X is digit)**.** Forexample **CH001**, **CH002**, **CH123**, etc.



*Figure 9. Input Challenger ID*

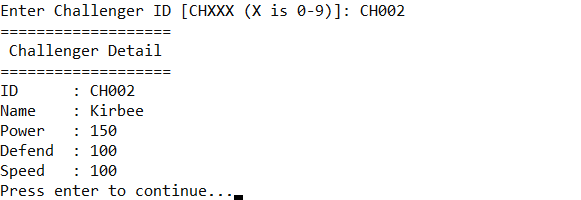
1. The program will **search** for the **specified challenger (case sensitive)**.

* If the **challenger id** is **not found** the program will **show** the **challenger ID not found** **message** and user **will be redirect back to menu page**.



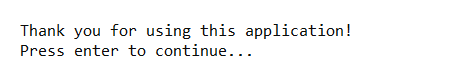
*Figure 10. Challenger not Found Message*

* If the **challenger id exists**, then:
  + The program will **remove** the **challenger data** from the **chaining hash table**
  + The program will **show** the **removed challenger detail**.



*Figure 11. Challenger Detail to Delete*

* If the user choose **menu 4** (**Exit**), then the program will **show** **exit** **message** and **exit**.



*Figure 12. Exit Message*

**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