# SIT102 – Introduction to Programming

## Answers for 5.2P Structs and Enums

Student Name: yizheng he
Student ID: 221411294

Question 1: Provide a **real-world example** of data that would be best managed by a struct of a C++ program. **Explain with code samples aligned with your given real-world example** how managing this kind of data in a struct will simplify the program code and facilitate the data structure.

A structure is excellent when it represents an entity in a program, it helps simplify the code by giving structure to its values and the structure helps simplify the code by keeping all the data needed organized and easy to read and find in our code. To take an example from real life, record a person's daily weight, age, name and gender.

```
Struct my_data
{
 string name;
 String gender;
 Int age;
 Double weight;
}
```

Question 2: Provide a **real- world example** of data you could manage with an enumeration of a C++ program. **Explain with code samples aligned with your given real-world example** how enumerations help make code easier to understand and work with.

Enumeration or enums are ways to name different data types and their value collectively without making separate functions or procedure it helps to make the code easier and is used to create different variables  access them and update them

```
enum knight_update_option
{
    UPDATE_NAME,
    UPDATE_AGE,
    UPDATE_BIRTH,
    UPDATE_GENDER,
    FINISH_UPDATE
};
```

Question 3: How does pass by reference help us work with data inside a struct? Explain how you can use this together with functions/procedures and parameters to organise the code associated with a program.

| |
|---|
| Pass by reference helps in updating the structs and the data inside them within a function or procedure created. It divides the large task into smaller tasks within the code in the main function & is pointer in reference used to point to the parameter in the functions and procedure<br>Instead of passing a value, it is possible to just create a reference to the data, which means that the value can be adjusted by not using a constant, which makes it possible to read or write the state. If you don't want it to change, you can add a constant in front of it. All this makes the program much cleaner and faster, especially for larger structures. |
| |
| |
| |
| |

End of questions (3)