HELLO!

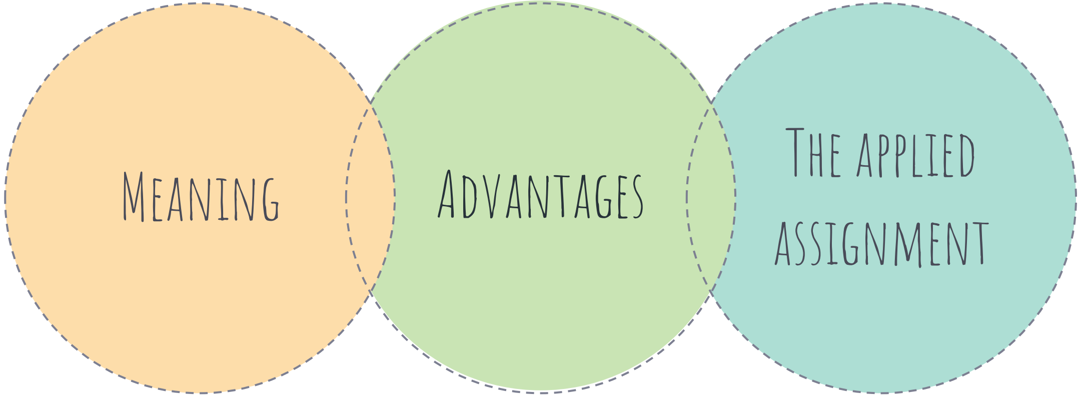
My name is Luc

I’m 22 years old and now I’m Final-year student at Hutech University

You can find me at letrantanluc2002@gmail.com

Topdown appoach

Topdown appoach



Meaning

Advantages

The applied assignment

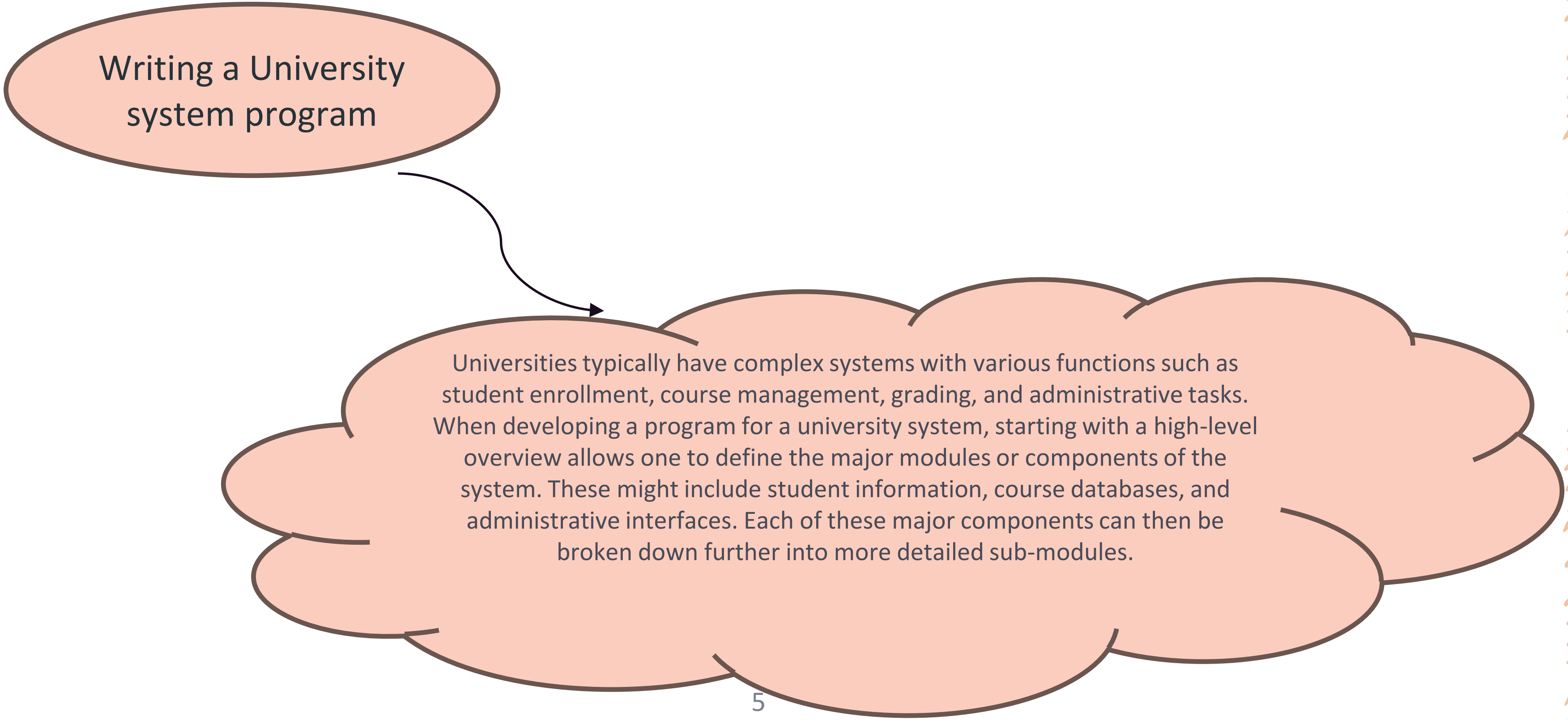
Meaning

In the top-down model, an overview of the system is formulated without going into detail for any part of it.

Each part of it then refined into more details, defining it in yet more details until the entire specification is detailed enough to validate the model.

If we glance at a haul as a full, it’s going to appear not possible as a result of it’s so complicated.

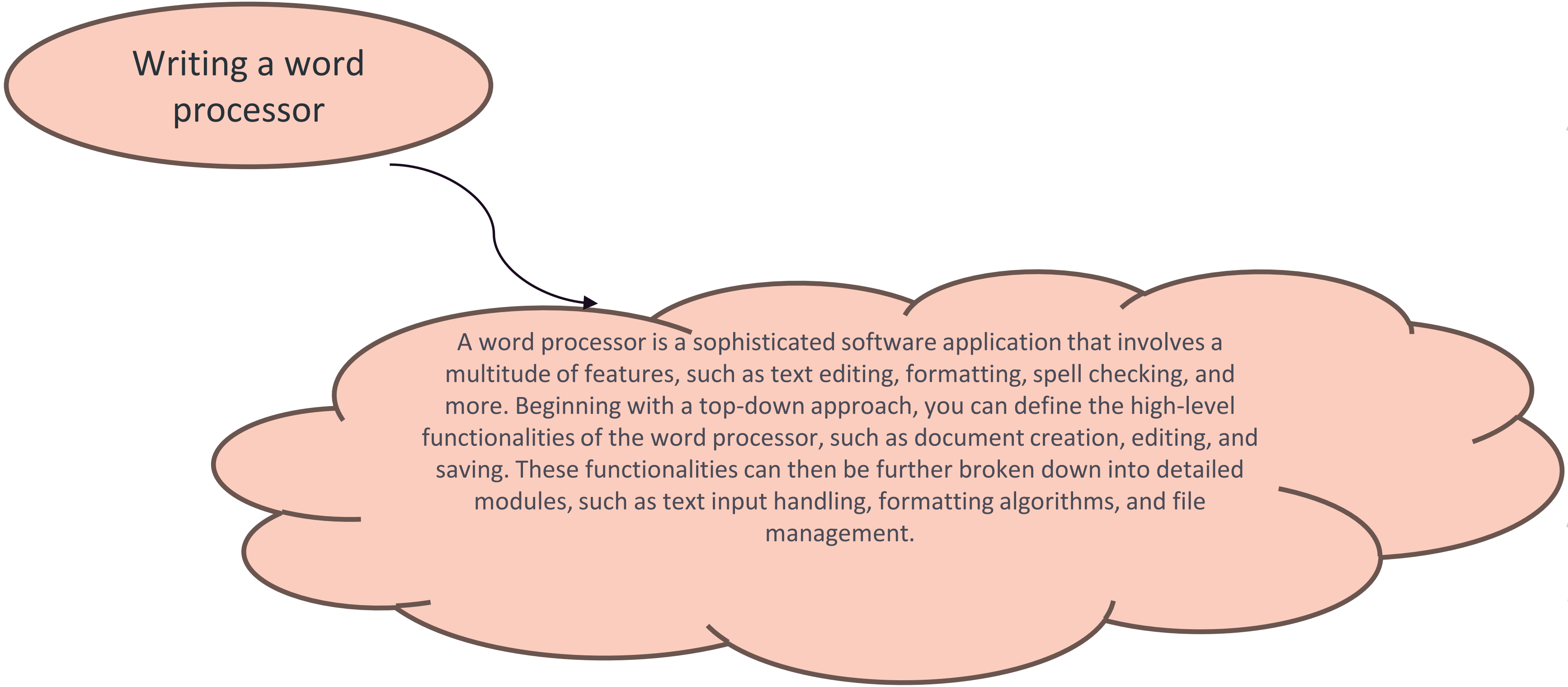
Example



Writing a University system program

Universities typically have complex systems with various functions such as student enrollment, course management, grading, and administrative tasks. When developing a program for a university system, starting with a high-level overview allows one to define the major modules or components of the system. These might include student information, course databases, and administrative interfaces. Each of these major components can then be broken down further into more detailed sub-modules.

Example



Writing a word processor

A word processor is a sophisticated software application that involves a multitude of features, such as text editing, formatting, spell checking, and more. Beginning with a top-down approach, you can define the high-level functionalities of the word processor, such as document creation, editing, and saving. These functionalities can then be further broken down into detailed modules, such as text input handling, formatting algorithms, and file management.

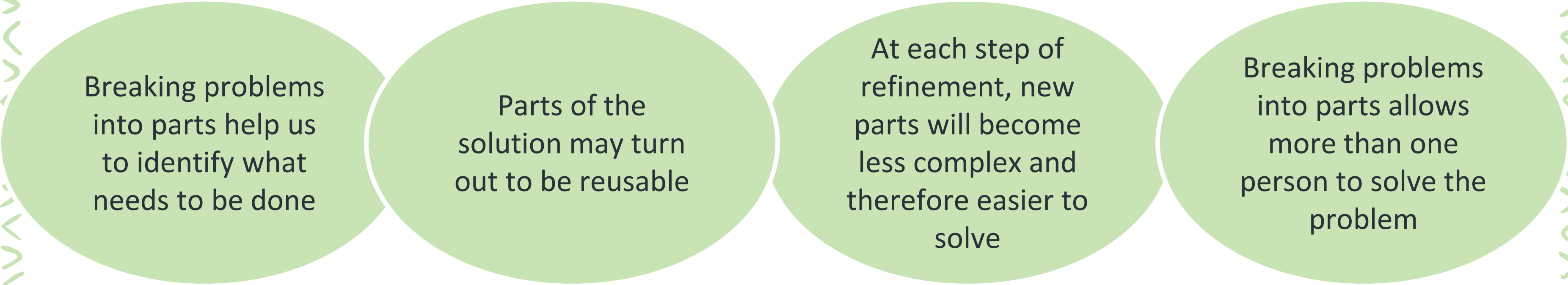
Meaning

We break the problem into parts

Then break the parts into parts soon and now each of parts will be easy to do.

Complicated issues may be resolved victimization high down style, conjointly referred to as Stepwise refinement where

Advantages



Breaking problems into parts help us to identify what needs to be done

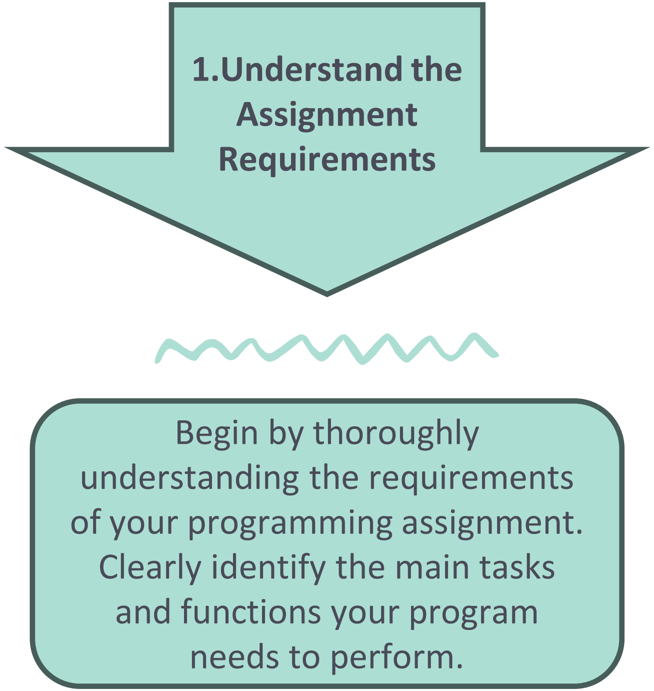
Parts of the solution may turn out to be reusable

At each step of refinement, new parts will become less complex and therefore easier to solve

Breaking problems into parts allows more than one person to solve the problem

The applied assignment

The applied assignment



1.Understand the Assignment Requirements

Begin by thoroughly understanding the requirements of your programming assignment. Clearly identify the main tasks and functions your program needs to perform.

The applied assignment

2.Create a High-Level Design:

Develop an overall design for your program. Define the main functions and key components that your program will include. This could be represented in a structural diagram or a high-level model.

The applied assignment

3.Divide the Program into Main Modules

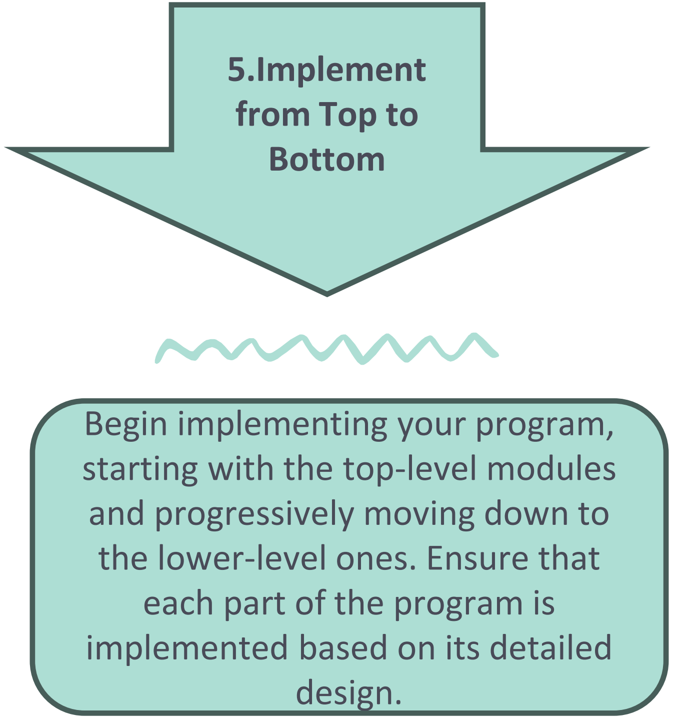
Break down your program into main modules based on the high-level design. Give each module a name and describe its function.

The applied assignment

4.Detail Each Module from Top to Bottom

Start detailing each module, beginning with the highest-level module. Identify and define the specific functions or methods within each module. This step involves breaking down each module into smaller functions or methods.

The applied assignment



5.Implement from Top to Bottom

Begin implementing your program, starting with the top-level modules and progressively moving down to the lower-level ones. Ensure that each part of the program is implemented based on its detailed design.

The applied assignment

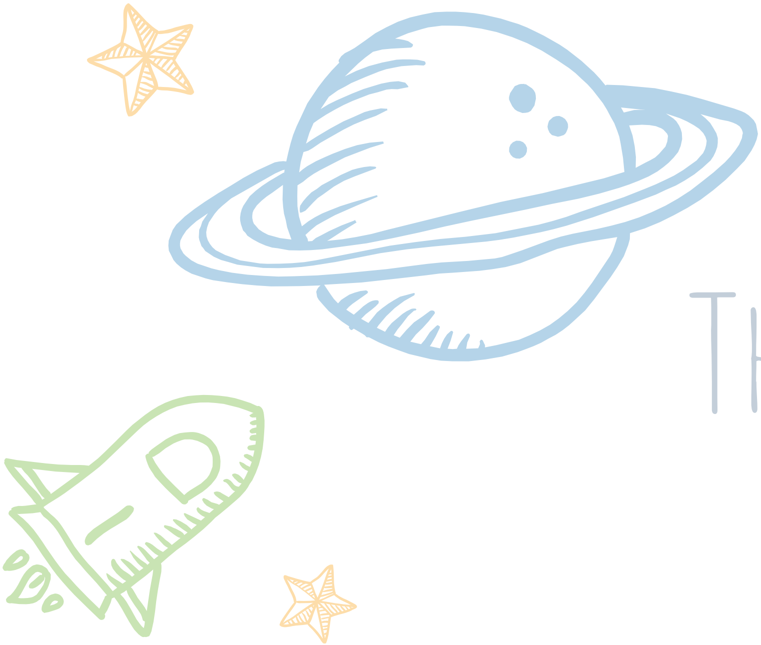
6.Test and Debug

Test each module individually and then integrate them to test the entire program. Debug and address any issues that arise during testing.

The applied assignment

7.Optimize and Finalize

Optimize your program and ensure that it operates efficiently. Make any necessary adjustments or improvements based on testing feedback.



Thank for watching