**ASSIGNMENT 2 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 1: Programming | | |
| **Submission date** |  | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Nguyen Ngoc Duy Hung | **Student ID** | BH01385 |
| **Class** | SE07101 | **Assessor name** | Nguyen Thanh Trieu |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P3 | P4 | P5 | P6 | M2 | M3 | M4 | D2 | D3 | D4 |
|  |  |  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Lecturer Signature:** | | |

Contents

[I. Introduction 4](#_Toc163684301)

[II. Body 4](#_Toc163684302)

[P3: Discuss what procedural, object-orientated and event-driven paradigms are; their characteristics and the relationship between them. 4](#_Toc163684303)

[ Procedural Paradigm: 4](#_Toc163684304)

[ Object Oriented Programming: 4](#_Toc163684305)

[ Event-Driven Paradigms 5](#_Toc163684306)

[ The relationship between the Procedural – OOP - Event-Driven Paradigms 5](#_Toc163684307)

[III. Conclusion. 7](#_Toc163684308)

[IV. Reference. 7](#_Toc163684309)

# Introduction

I am a water billing application programmer. Now I will program a water billing application for users to use conveniently and easily.

# Body

## P3: Discuss what procedural, object-orientated and event-driven paradigms are; their characteristics and the relationship between them.

Let’s find out what procedural, object-orientated and event-driven paradigms are and their characteristics and the relationship between them.

### Procedural Paradigm:

Procedural Paradigm is a programming model that uses a linear or top-down approach. It relies on procedures or subroutines to perform calculations.

Example: You want to make a cake you have to do 3 steps respectively: Mix the flour - Pour into mold -Put in the oven.

In short, it is suitable for small things and doing them in a certain order.

### Object Oriented Programming:

OOP is a programming method based on the concept of classes and objects. OOP focuses on manipulating objects rather than the logic to manipulate them, making code more manageable, reusable, and maintainable.

The goal of OOP is to optimize source code management, help increase reusability, and most importantly, help summarize procedures with known properties by objects.

4 basic characteristics of OOP:

* Encapsulation allows hiding information and internal processing properties of objects.
* Inheritance allows building a new class, inheriting and reusing properties and methods based on a previously existing class.
* Polymorphism in OOP programming allows different objects to execute the same function in different ways.
* Abstraction helps eliminate complex, unnecessary things from an object and focus only on what is core and important.

### Event-Driven Paradigms

Event-driven programming is a school of programming in which entities communicate indirectly with each other by sending messages to each other through middleware. Messages are stored in a queue before being processed by consumers.

Event-driven programming allows the development of responsive and interactive applications that can handle user input and external events.

### The relationship between the Procedural – OOP - Event-Driven Paradigms

Procedural, object-oriented, and event-driven models do not cancel each other out but they can be used together to aid each other. In fact, many modern programming languages, such as C#, Java, and Python, support multiple paradigms.

* OO with Event Driven programming: objects make it easier to represent the environment and nuance changes to that environment.
* Procedural and Event-Driven: Procedural programming is about following an established sequence of steps, event-driven programming is about responding to user or system actions. They can be used together in the same program, by dictating the events to which the event driver responds.
* Object-Oriented and Event-Driven: Object-oriented programming can be combined with event-oriented programming by defining main classes for objects that handle events. Those objects encapsulate event handlers and related functions.

In short, Procedure Oriented and Event Oriented describe the general workflow of an application or decision making, while Object Oriented describes more of the structure of decision making. But these models can be used together and are often interconnected, not mutually exclusive, although they still have distinct characteristics.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Procedural | OOP | Event-Driven |
| Advantages | - It simplifies the code and makes it easy to understand  - Implementing algorithms is easy | - It allows breaking the program into bit-sized problems that can be solved easily  - It promises greater programmer productivity, better quality of software, and lesser maintenance cost | - It provides flexibility as programs can be altered easily  - It is suitable for graphical interfaces |
| Disavantages | - The data is vulnerable  - For complex programs, procedural programming can be less efficient and harder to manage | - Programmers need to have excellent designing skill and programming skill along with proper planning  - OOP is not a universal language and can't be applied everywhere | - The flow of the program is usually less logical and obvious.  - Errors can be more difficult to spot than with simpler, procedural programs. |
| Similar | - Both procedural and event-driven models describe the overall workflow of the application  - All paradigms can be used together in a single program  - Can support code reuse | | |
| Differences | - Describes a sequential working process of logic.  - Must be done in order  - Does not need to exist in a context outside of logic | - OOP works based on the model.  - It describes the decision making. | - The system issues notifications or responses are given from other systems.  - Processing is not synchronized with each other  - Usually focuses mainly on events |

# Conclusion.

My presentation on Procedural, OOP, Event-Driven Paradigms highlights the unique characteristics of each Paradigms.

**And here is my link led to my code:** <https://github.com/MimiRamie/tinh-tien-nc>

# Reference.

Pham, H.D. (no date) *Sự khác biệt giữa lập trình hướng thủ tục (POP) và lập trình hướng đối tượng (OOP)*. <https://viblo.asia/p/su-khac-biet-giua-lap-trinh-huong-thu-tuc-pop-va-lap-trinh-huong-doi-tuong-oop-bWrZnxOm5xw>.

Bui, D. (2024) *Procedural programming và Object-Oriented programming - Viblo*. <https://viblo.asia/p/001-procedural-programming-va-object-oriented-programming-3P0lP3nvZox>.

*Event-driven programming là gì? Tính ứng dụng trong quá trình phát triển phần mềm* (no date). <https://stringee.com/vi/blog/post/Event-driven-programming-la-gi>.

*What are the relationships among procedural, object oriented and event driven paradigms?* (no date). <https://stackoverflow.com/questions/74426618/what-are-the-relationships-among-procedural-object-oriented-and-event-driven-pa>.

‌