❀ Programming For Everybody ❀

1. **Exploiting the Capability of Computer Science in Afghanistan and global:**
   - ✤ This module explores the potential and capability of computer science in the global and Afghan context.
   - ✤ The module examines the latest advancements and developments in computer science.
   - ✤ By studying the advancements, students will gain an understanding of the important role computer science plays in shaping the world and future.

2. **Exploring the Intersection of Technology and Computer Science and the Definition of Related Terms:**
   - ✤ Key concepts and terms of computer science will be explored in this module.
   - ✤ Topics covered include technology, science, engineering, computing, and more.
   - ✤ Understanding these terms is essential for understanding the principles of computer science and the role of technology in our lives.
   - ✤ This module is suitable for both beginners and experienced professionals.
   - ✤ Provides a comprehensive understanding of the fundamental terms in computer science.

3. **A Computer Odyssey (From Past to Present):**
   - ✤ This module explores the history of computing.
   - ✤ The module covers the major milestones and innovations in the field of computing.
   - ✤ The module covers the history of computing from its early beginnings to the present day.
   - ✤ This module explores a comprehensive understanding of computer parts.
   - ✤ The module includes a comprehensive guide to programming languages and their evaluation throughout history.

4. **Understanding our brain to learn better:**
   - ✤ Welcome to module on Understanding our Brain to Learn Better
   - ✤ Brains are complex and powerful organs critical to learning and memory
   - ✤ This module explores latest research and findings on brain structure and function
   - ✤ Focuses on how this knowledge can be used to improve learning and memory skills

- Suitable for students looking to optimize study habits or professionals looking to enhance training and development
- Module provides valuable insights and practical strategies for learning more effectively.

5. **Introduction to Python:**
    - Overview of the programming language, it's history, evaluation, and usage.
    - Setting up the development environment and installing required tools.
    - Writing the first program in Python and understanding the basic syntax.
    - Advises and tips before starting with **Programming.**

2. **Primitive Types and Variables:**
    - Overview of different data types in Python (e.g. integers, floats, strings, lists, etc.)
    - Working with variables and assigning values to them.
    - Type casting and type coercion.

3. **Operators and Expressions:**
    - Overview of different types of operators (e.g. arithmetic, comparison, logical, etc.)
    - Writing expressions and evaluating them.
    - Operator precedence and associativity.

4. **Control Flow:**
    - Conditional statements (if, elif, else).
    - Loops (for, while).
    - Using control flow statements to make decisions and control the flow of a program.

5. **Functions:**
    - Overview of functions and their usage.
    - Defining and calling functions.
    - Function arguments and return values.
    - Scope and lifetime of variables.

6. **Data Structures:**
    - Overview of the term data structure.
    - Explaining deferent data structure in python (lists, tuples, sets, dictionaries)
    - Exploring list comprehensions.

7. **Modules and Packages:**
    - Overview of modules and packages in Python.
    - Importing and using existing modules.
    - Creating and distributing your own modules and packages.

7. **File Input/Output:**

- Reading and writing files in Python.
- Understanding the different modes of file operations (e.g. read, write, append, etc.)
- Working with CSV, JSON and XML file formats.

8. **Exception Handling:**

- Overview of exceptions and errors in Python.
- Handling exceptions using try-except blocks.
- Raising exceptions in your code.

9. **Object-Oriented Programming:**

- Overview of OOP concepts (classes, objects, inheritance, polymorphism, etc.)
- Defining and using classes and objects in Python.
- Overriding methods and using inheritance.

11. **Advanced Topics:**

- Introduction **lambda, map and filter** functions .
- Overview of advanced topics in Python such as decorators, generators, regular expressions, etc.
- Using these concepts to write more efficient and elegant code.

12. **Introduction to Git and Github:**

- Overview of Git & Github.
- Creating profiles.
- Introduction to basic Git & Github (project creating and integration).

13. **Milestone Projects**

- Overview of some well-known application created using python.
- Overview of what kind of project can be created with python.
- Some real-time projects on diverse areas such as:
    - Making Website with Python.
    - Creating GUI based Desktop application.
    - Building a Data Science Related Project and Visualisation.
    - Etc…

14. **Conclusion and Next Steps:**

- Recap of the course and its objectives.
- Discussion of future prospects and next steps for students.
- Suggesting additional resources and exercises for further learning.