

Postgraduate Diploma Programming





Postgraduate Diploma Programming

- » Modality: online
- » Duration: 6 months
- » Certificate: TECH Technological University
- » Dedication: 16h/week
- » Schedule: at your own pace
- » Exams: online

Website: www.techtute.com/us/information-technology/postgraduate-diploma/postgraduate-diploma-programming

Index

01

Introduction

p. 4

02

Objectives

p. 8

03

Structure and Content

p. 12

04

Methodology

p. 18

05

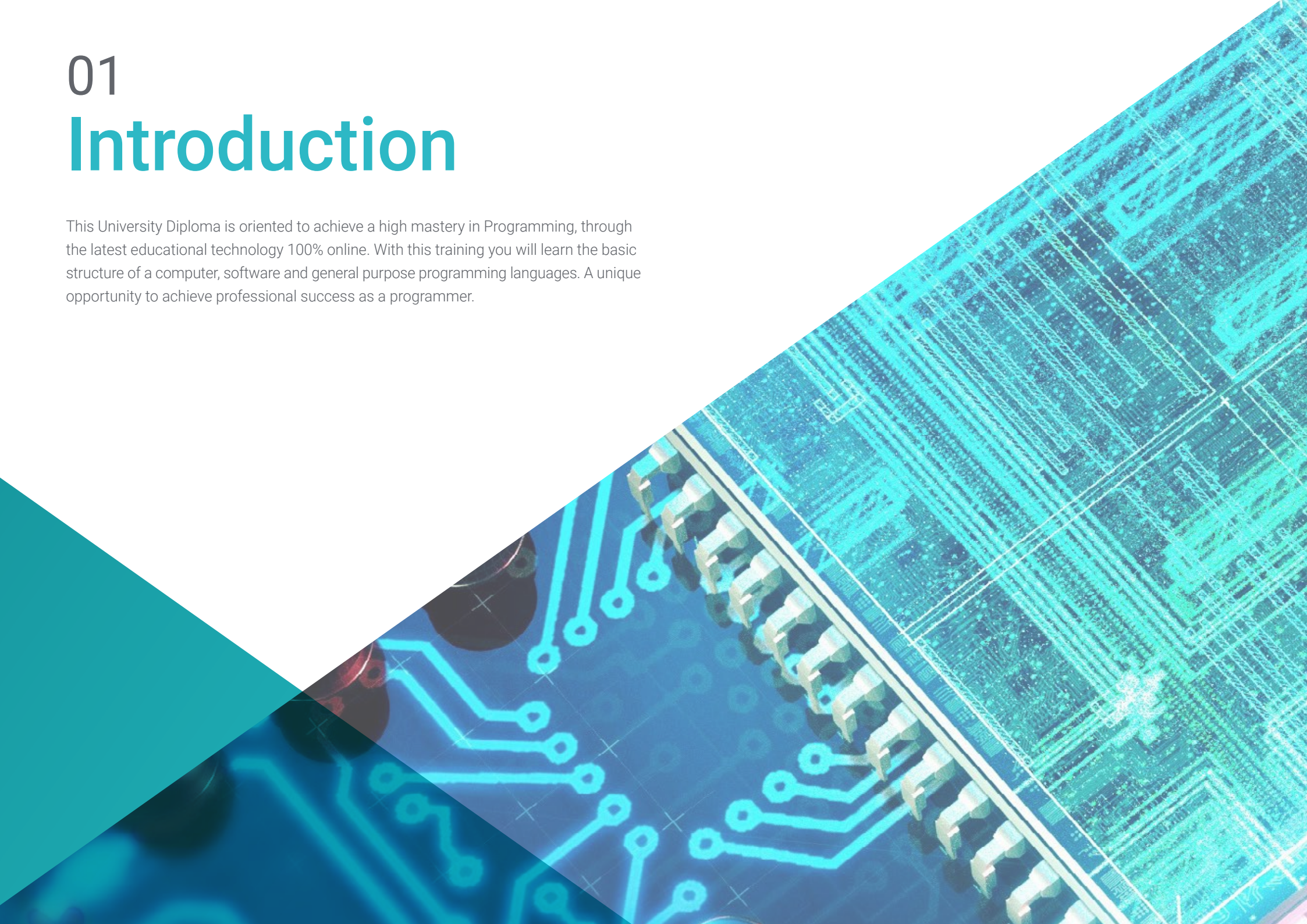
Certificate

p. 26

01

Introduction

This University Diploma is oriented to achieve a high mastery in Programming, through the latest educational technology 100% online. With this training you will learn the basic structure of a computer, software and general purpose programming languages. A unique opportunity to achieve professional success as a programmer.



“

This University Expert will allow you up to date your knowledge about programming in a practical way, 100% online, without renouncing to the maximum academic rigor”

This program is aimed at those interested in attaining a higher level of knowledge of Programming. The main objective of this Postgraduate Diploma is for students to specialize their knowledge in simulated work environments and conditions in a rigorous and realistic manner so that they can later apply it in the real world.

This program will prepare scientifically and technologically, as well as to develop the professional practice of software engineering, with a transversal and versatile approach adapted to the new technologies and innovations in this field. Students will gain extensive knowledge Programming from professionals in the field.

The professional must Seize the opportunity to take this program program in a 100% online format, without having to give up obligations, which will make it easier to continue studying. Up to date your knowledge and get a Postgraduate Diploma to continue growing personally and professionally.

This **Postgraduate Diploma in Programming** contains the most complete and up-to-date program on the market. The most important features include:

- ◆ Development of 100 simulated scenarios presented by experts in Programming
- ◆ The graphic, schematic and practical contents with which they are conceived provide scientific and practical information on Programming
- ◆ News on the latest developments in Information Systems
- ◆ It contains practical exercises where the self-assessment process can be carried out to improve learning
- ◆ Interactive learning system based on the case method and its application to real practice
- ◆ All of this will be complemented by theoretical lessons, questions to the expert, debate forums on controversial topics, and individual reflection assignments
- ◆ Content that is accessible from any fixed or portable device with an internet connection



This program will allow you to enhance your skills and update your knowledge in Programming"

“

Learn about Information Systems with this intensive program , from the comfort of your home”

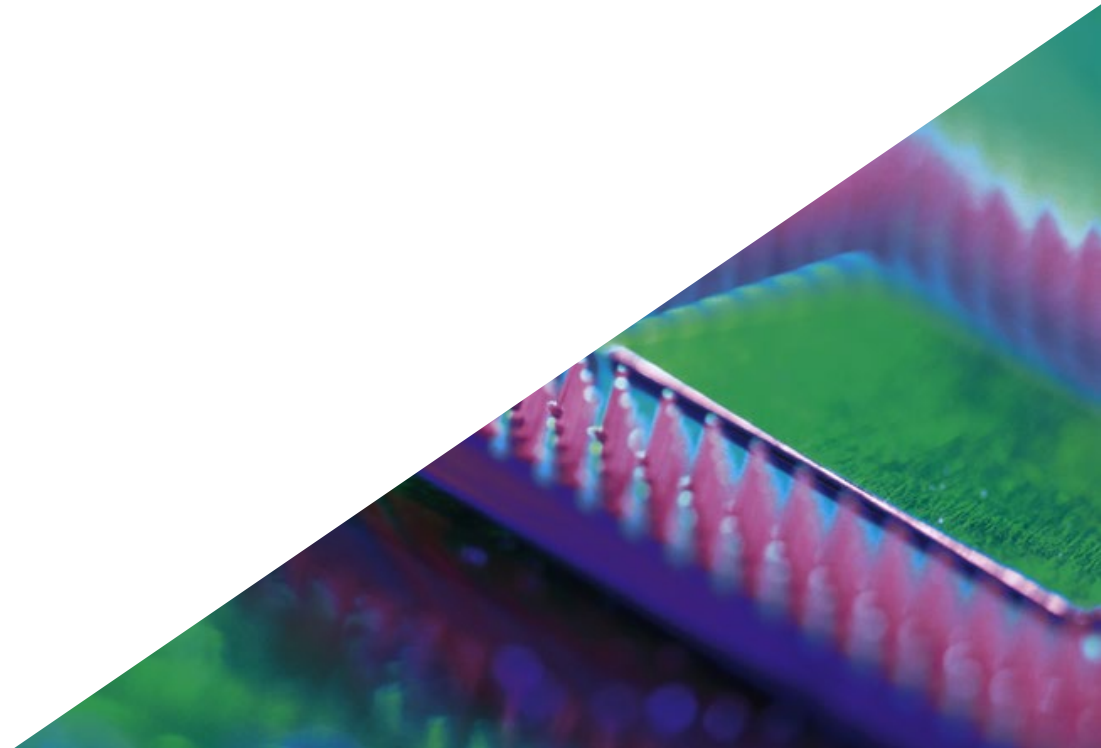
Take advantage of the latest educational technology to update on Programming from the confort of your home.

Learn about the latest techniques in Information Systems from experts in the field.

The program's teaching staff includes professionals from the sector who contribute their work experience to this training program, as well as renowned specialists from leading societies and prestigious universities.

Its multimedia content, developed with the latest educational technology, will allow the professional a situated and contextual learning, that is, a simulated environment that will provide an immersive training programmed to train in real situations.

The design of this program focuses on Problem-Based Learning, in which the professional will have to try to solve the different professional practice situations that will arise throughout the academic course. For this purpose, the student will be assisted by an innovative interactive video system created by renowned experts.



02

Objectives

The objective of this program is to provide IT professionals with the knowledge and skills necessary to carry out their activity using the most advanced protocols and techniques of the moment. Through a work approach that is totally adaptable to the students, this Postgraduate Diploma will progressively lead them to acquire the competencies that will propel them to a higher professional level.



“

You will delve into the computer's field and computer structure; essential subjects for any software developer"



Specific Objectives

Module 1. Programming Fundamentals

- ◆ Understand the basic structure of a computer, software and general-purpose programming languages
- ◆ Learn to design and interpret algorithms, which are the necessary basis for developing computer programs
- ◆ Understand the essential elements of a computer program, such as the different types of data, operators, expressions, statements, I/O and control statements.
- ◆ Understand the different data structures available in general purpose programming languages, both static and dynamic, as well as to acquire the essential knowledge for file handling
- ◆ Know the different testing techniques in computer programs and the importance of generating good documentation together with good source code
- ◆ Learn the basic concepts of the C++ programming language, one of the most widely used languages in the world

Module 2. Data Structure

- ◆ Learn the fundamentals of C++ programming, including classes, variables, conditional expressions and objects
- ◆ Understand abstract data types, linear data structure types, simple and complex hierarchical data structures, as well as their implementation in C++
- ◆ Understand the operation of advanced data structures other than the usual ones
- ◆ Know the theory and practice related to the use of priority heaps and queues
- ◆ Learn how Hash tables work as abstract data types and functions
- ◆ Understand graph theory, as well as advanced graph algorithms and concepts

Module 3. Advanced Programming

- ◆ In-depth knowledge of programming, especially C++ relates to object-oriented programming, and the different types of relationships between existing classes
- ◆ Know the different design patterns for object-oriented problems
- ◆ Learn about event-driven programming and the development of user interfaces with Qt
- ◆ Acquire the essential knowledge of Concurrent Programming, processes and threads
- ◆ Learn how to manage the use of threads and synchronization, as well as the resolution of common problems within Concurrent Programming
- ◆ Understand the importance of documentation and testing in software development

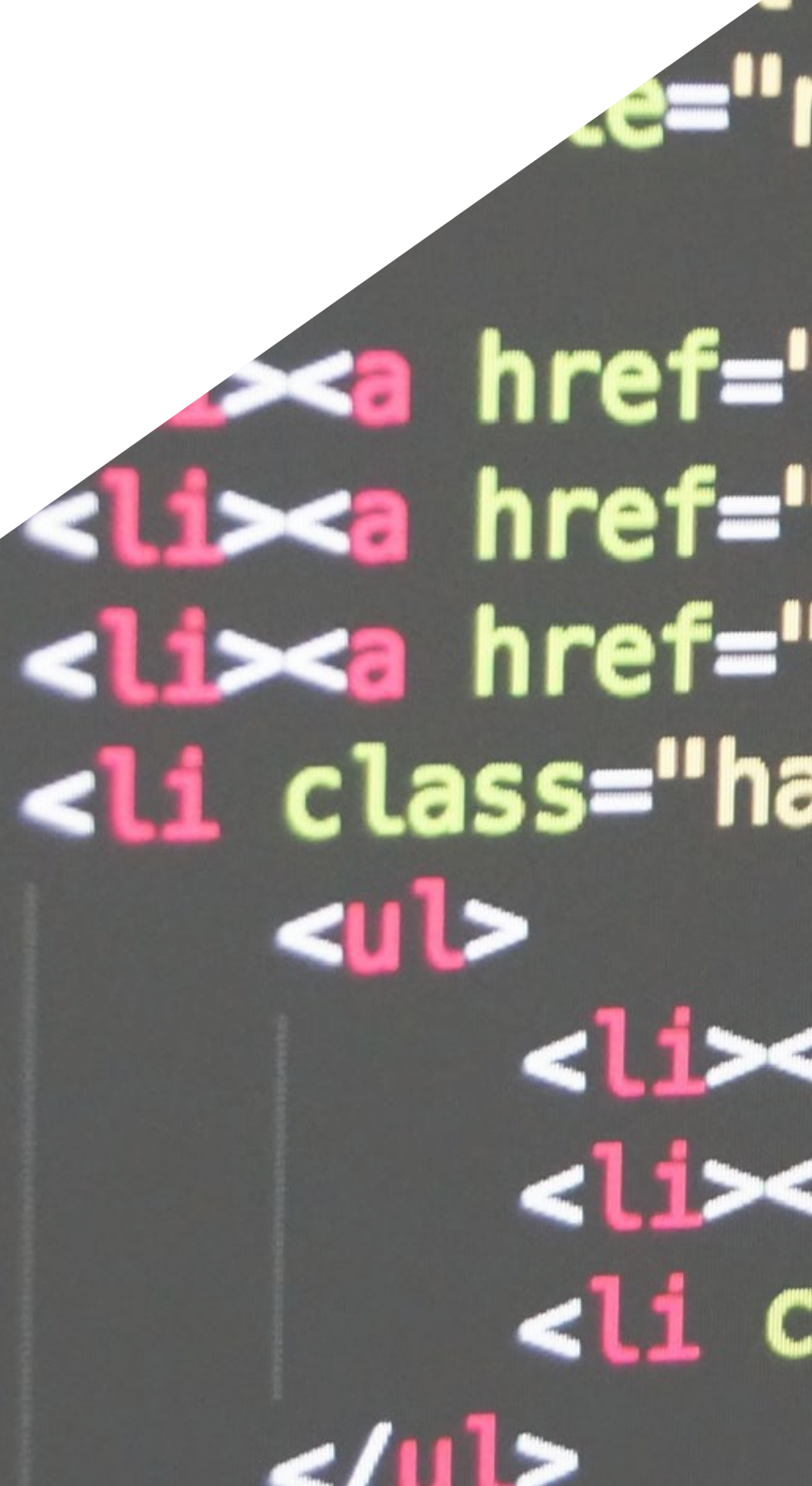
Module 4. Development of Web Applications

- ◆ Know the characteristics of the HTML markup language and its use in web creation together with CSS style sheets.
- ◆ Learn to use the browser-oriented programming language JavaScript, and some of its main features
- ◆ Understand the concepts of Component Oriented Programming and Component Architecture
- ◆ Learn how to use the Bootstrap front-end framework for website design
- ◆ Understand the structure of the controller view model in the development of dynamic web sites
- ◆ Know the service-oriented architecture and the basics of the HTTP protocol

03

Structure and Content

The structure of the contents has been designed by a team of IT engineering professionals, aware of the relevance of current preparation in order to delve into this area of knowledge, in order to humanistically enrich the students and raise the level of knowledge in Information Systems through the latest educational technologies available.



```
navigation">
```

```
'index.html">Home</a></li>
```

```
'home-events.html">Home Events
```

```
'multi-col-menu.html">Multi
```

```
as-children"> <a href
```

```
<a href="tail
```

```
<a href="i
```

```
class="
```

“

This Postgraduate Diploma in Programming contains the most complete and up-to-date program on the market”

Module 1. Programming Fundamentals

- 1.1. Introduction to Programming
 - 1.1.1. Basic Structure of a Computer
 - 1.1.2. Software
 - 1.1.3. Video Game
 - 1.1.4. Life Cycle of a Software Application
- 1.2. Algorithm Design
 - 1.2.1. Problem Solving
 - 1.2.2. Descriptive Techniques
 - 1.2.3. Algorithm Elements and Structure
- 1.3. Elements of a Program
 - 1.3.1. C++ Origin and Features
 - 1.3.2. Development Environment
 - 1.3.3. Concept of Program
 - 1.3.4. Types of Fundamental Data
 - 1.3.5. Operators
 - 1.3.6. Expressions
 - 1.3.7. Statements
 - 1.3.8. Data Input and Output
- 1.4. Control Sentences
 - 1.4.1. Statements
 - 1.4.2. Branches
 - 1.4.3. Loops
- 1.5. Abstraction and Modularity: Functions
 - 1.5.1. Modular Design
 - 1.5.2. Concept of Function and Utility
 - 1.5.3. Definition of a Function
 - 1.5.4. Execution Flow in a Function Call
 - 1.5.5. Function Prototypes
 - 1.5.6. Results Return
 - 1.5.7. Calling a Function: Parameters
 - 1.5.8. Passing Parameters by Reference and by Value
 - 1.5.9. Scope Identifier
- 1.6. Static Data Structures
 - 1.6.1. Arrays
 - 1.6.2. Matrices, Polyhedra
 - 1.6.3. Searching and Sorting
 - 1.6.4. I/O Functions for Chains
 - 1.6.5. Structures
 - 1.6.6. New Types of Data
- 1.7. Dynamic Data Structures: Pointers
 - 1.7.1. Definition of pointer
 - 1.7.2. Pointer Operators and Operations
 - 1.7.3. Pointer Arrays
 - 1.7.4. Pointers and Arrays
 - 1.7.5. Chain Pointers
 - 1.7.6. Structure Pointers
 - 1.7.7. Multiple Indirection
 - 1.7.8. Function Pointers
 - 1.7.9. Function, Structure and Array Passing as Function Parameters
- 1.8. Files
 - 1.8.1. Basic Concepts
 - 1.8.2. File Operations
 - 1.8.3. Types of Files
 - 1.8.4. File Organization
 - 1.8.5. Introduction to C++ Files
 - 1.8.6. Managing Files
- 1.9. Recursion
 - 1.9.1. Definition of Recursion
 - 1.9.2. Types of Recursion
 - 1.9.3. Advantages and Disadvantages
 - 1.9.4. Considerations
 - 1.9.5. Recursive-Iterative Conversion
 - 1.9.6. Recursion Stack

- 1.10. Testing and Documentation
 - 1.10.1. Program Testing
 - 1.10.2. White Box Testing
 - 1.10.3. Black Box Testing
 - 1.10.4. Testing Tools
 - 1.10.5. Program Documentation

Module 2. Data Structure

- 2.1. Introduction to C++ Programming
 - 2.1.1. Classes, Constructors, Methods and Attributes
 - 2.1.2. Variables
 - 2.1.3. Conditional Expressions and Loops
 - 2.1.4. Objects
- 2.2. Abstract Data Types (ADT)
 - 2.2.1. Types of Data
 - 2.2.2. Basic Structures and TADs
 - 2.2.3. Vectors and Arrays
- 2.3. Linear data Structures
 - 2.3.1. ADT Ready Definition
 - 2.3.2. Linked and Doubly Linked Lists
 - 2.3.3. Sorted Lists
 - 2.3.4. Lists in C++
 - 2.3.5. ADT Stack
 - 2.3.6. ADT Queue
 - 2.3.7. Stack and Queue in C++
- 2.4. Hierarchical Data Structures
 - 2.4.1. ADT Tree
 - 2.4.2. Paths
 - 2.4.3. N-Ary Trees
 - 2.4.4. Binary Trees
 - 2.4.5. Binary Search Trees
- 2.5. Hierarchical Data Structures: Complex Trees
 - 2.5.1. Perfectly Balanced or Minimum Height Trees
 - 2.5.2. Multipath Trees
 - 2.5.3. Bibliographical References
- 2.6. Priority Mounds and Queue
 - 2.6.1. ADT Heaps
 - 2.6.2. ADT Priority Queues
- 2.7. Hash Tables
 - 2.7.1. TAD Hash Table
 - 2.7.2. Hash Functions
 - 2.7.3. Hash Function in Hash Tables
 - 2.7.4. Redispersion
 - 2.7.5. Open Hash Tables
- 2.8. Graphs
 - 2.8.1. TAD Graphs
 - 2.8.2. Types of Graphs
 - 2.8.3. Graphical Representation and Basic Operations
 - 2.8.4. Graph Design
- 2.9. Algorithms and Advanced Graph Concepts
 - 2.9.1. Problems about Graphs
 - 2.9.2. Path Algorithms
 - 2.9.3. Search or Path Algorithms
 - 2.9.4. Other Algorithms
- 2.10. Other Data Structures
 - 2.10.1. Sets
 - 2.10.2. Parallel Arrays
 - 2.10.3. Symbol Tables
 - 2.10.4. Tries

Module 3. Advanced Programming

- 3.1. Introduction to Object-Oriented Programming
 - 3.1.1. Introduction to Object-Oriented Programming
 - 3.1.2. Class Design
 - 3.1.3. Introduction to UML for Problem Modeling
- 3.2. Relationships Between Classes
 - 3.2.1. Abstraction and Inheritance
 - 3.2.2. Advanced Inheritance Concepts
 - 3.2.3. Polymorphism
 - 3.2.4. Composition and Aggregation
- 3.3. Introduction to Design Patterns for Object-Oriented Problems
 - 3.3.1. What are Design Patterns?
 - 3.3.2. Factory Pattern
 - 3.3.3. Singleton Pattern
 - 3.3.4. Observer Pattern
 - 3.3.5. Composite Pattern
- 3.4. Exceptions
 - 3.4.1. What are Exceptions?
 - 3.4.2. Exception Catching and Handling
 - 3.4.3. Throwing Exceptions
 - 3.4.4. Exception Creation
- 3.5. User Interfaces
 - 3.5.1. Introduction to Qt
 - 3.5.2. Positioning
 - 3.5.3. What Are Events?
 - 3.5.4. Events: Definition and Catching
 - 3.5.5. User Interface Development
- 3.6. Introduction to Concurrent Programming
 - 3.6.1. Introduction to Concurrent Programming
 - 3.6.2. The Concept of Process and Thread
 - 3.6.3. Interaction Between Processes or Threads
 - 3.6.4. Threads in C++
 - 3.6.5. Advantages and Disadvantages of Concurrent Programming

- 3.7. Thread Management and Synchronization
 - 3.7.1. Life Cycle of a Thread
 - 3.7.2. Thread Class
 - 3.7.3. Thread Planning
 - 3.7.4. Thread Groups
 - 3.7.5. Daemon Threads
 - 3.7.6. Synchronization
 - 3.7.7. Locking Mechanisms
 - 3.7.8. Communication Mechanisms
 - 3.7.9. Monitors
- 3.8. Common Problems in Concurrent Programming
 - 3.8.1. The Problem of Consuming Producers
 - 3.8.2. The Problem of Readers and Writers
 - 3.8.3. The Problem of the Philosophers' Dinner Party
- 3.9. Software Documentation and Testing
 - 3.9.1. Why is it Important to Document Software?
 - 3.9.2. Design Documentation
 - 3.9.3. Documentation Tool Use
- 3.10. Software Testing
 - 3.10.1. Introduction to Software Testing
 - 3.10.2. Types of Tests
 - 3.10.3. Unit Test
 - 3.10.4. Integration Test
 - 3.10.5. Validation Test
 - 3.10.6. System Test

Module 4. Development of Web Applications

- 4.1. HTML5 Markup Languages
 - 4.1.1. HTML Basics
 - 4.1.2. New HTML 5 Elements
 - 4.1.3. Forms: New Controls
- 4.2. Introduction to CSS Style Sheets
 - 4.2.1. First Steps with CSS
 - 4.2.2. Introduction to CSS3

```
form.querySelectorAll('input[type=checkbox][name="' + b.el.name + "']")
index(b.el)) {
d.filter(":checked").length;
>= b.arg || g.minChecked.replace("{count}", b.arg)

function(a) {
!= a.val ? a.val.length <= a.arg || g.maxSelected.replace("{count}", a.arg)

function(a) {
!= a.val && a.val.length >= a.arg || g.minSelected.replace("{count}", a.arg)

b) {
is.form.querySelectorAll('input[type=radio][name="' + b.name + "']").filter("

c

(a, b) {
tions.custom[a.arg],
RegExp(c.pattern);
t(a.val) || c.errorMessage

(a) {
.arg

his.options = a.extend(!0, {}, h, c), this.form = h, this.str = {}, this
```

- 4.3. Browser Scripting Language: JavaScript
 - 4.3.1. JavaScript Basics
 - 4.3.2. DOM
 - 4.3.3. Events
 - 4.3.4. JQuery
 - 4.3.5. Ajax
- 4.4. Concept of Component-Oriented Programming
 - 4.4.1. Context
 - 4.4.2. Components and Interfaces
 - 4.4.3. States of a Component
- 4.5. Component Architecture
 - 4.5.1. Current Architectures
 - 4.5.2. Component Integration and Deployment
- 4.6. Framework Front-End: Bootstrap
 - 4.6.1. Grid Design
 - 4.6.2. Forms
 - 4.6.3. Components
- 4.7. Model View Controller
 - 4.7.1. Web Development Methods
 - 4.7.2. Design Pattern: MVC
- 4.8. Information Grid Technologies
 - 4.8.1. Increased Computing Resources
 - 4.8.2. Concept of Grid Technology
- 4.9. Service-Oriented Architecture
 - 4.9.1. SOA and Web Services
 - 4.9.2. Topology of a Web Service
 - 4.9.3. Platforms for Web Services
- 4.10. HTTP Protocol
 - 4.10.1. Messages
 - 4.10.2. Persistent Sessions
 - 4.10.3. Cryptographic System
 - 4.10.4. HTTP Protocol Operation

04

Methodology

This academic program offers students a different way of learning. Our methodology uses a cyclical learning approach: **Relearning**.

This teaching system is used, for example, in the most prestigious medical schools in the world, and major publications such as the **New England Journal of Medicine** have considered it to be one of the most effective.



“

Discover Relearning, a system that abandons conventional linear learning, to take you through cyclical teaching systems: a way of learning that has proven to be extremely effective, especially in subjects that require memorization"

Case Study to contextualize all content

Our program offers a revolutionary approach to developing skills and knowledge. Our goal is to strengthen skills in a changing, competitive, and highly demanding environment.

“

At TECH, you will experience a learning methodology that is shaking the foundations of traditional universities around the world”



You will have access to a learning system based on repetition, with natural and progressive teaching throughout the entire syllabus.



The student will learn to solve complex situations in real business environments through collaborative activities and real cases.

A learning method that is different and innovative

This TECH program is an intensive educational program, created from scratch, which presents the most demanding challenges and decisions in this field, both nationally and internationally. This methodology promotes personal and professional growth, representing a significant step towards success. The case method, a technique that lays the foundation for this content, ensures that the most current economic, social and professional reality is taken into account.

“*Our program prepares you to face new challenges in uncertain environments and achieve success in your career”*

The case method has been the most widely used learning system among the world's leading Information Technology schools for as long as they have existed. The case method was developed in 1912 so that law students would not only learn the law based on theoretical content. It consisted of presenting students with real-life, complex situations for them to make informed decisions and value judgments on how to resolve them. In 1924, Harvard adopted it as a standard teaching method.

What should a professional do in a given situation? This is the question that you are presented with in the case method, an action-oriented learning method. Throughout the course, students will be presented with multiple real cases. They will have to combine all their knowledge and research, and argue and defend their ideas and decisions.

Relearning Methodology

TECH effectively combines the Case Study methodology with a 100% online learning system based on repetition, which combines different teaching elements in each lesson.

We enhance the Case Study with the best 100% online teaching method: Relearning.

In 2019, we obtained the best learning results of all online universities in the world.

At TECH you will learn using a cutting-edge methodology designed to train the executives of the future. This method, at the forefront of international teaching, is called Relearning.

Our university is the only one in the world authorized to employ this successful method. In 2019, we managed to improve our students' overall satisfaction levels (teaching quality, quality of materials, course structure, objectives...) based on the best online university indicators.



In our program, learning is not a linear process, but rather a spiral (learn, unlearn, forget, and re-learn). Therefore, we combine each of these elements concentrically.

This methodology has trained more than 650,000 university graduates with unprecedented success in fields as diverse as biochemistry, genetics, surgery, international law, management skills, sports science, philosophy, law, engineering, journalism, history, and financial markets and instruments. All this in a highly demanding environment, where the students have a strong socio-economic profile and an average age of 43.5 years.

Relearning will allow you to learn with less effort and better performance, involving you more in your training, developing a critical mindset, defending arguments, and contrasting opinions: a direct equation for success.

From the latest scientific evidence in the field of neuroscience, not only do we know how to organize information, ideas, images and memories, but we know that the place and context where we have learned something is fundamental for us to be able to remember it and store it in the hippocampus, to retain it in our long-term memory.

In this way, and in what is called neurocognitive context-dependent e-learning, the different elements in our program are connected to the context where the individual carries out their professional activity.



This program offers the best educational material, prepared with professionals in mind:



Study Material

All teaching material is produced by the specialists who teach the course, specifically for the course, so that the teaching content is highly specific and precise.

These contents are then applied to the audiovisual format, to create the TECH online working method. All this, with the latest techniques that offer high quality pieces in each and every one of the materials that are made available to the student.



Classes

There is scientific evidence suggesting that observing third-party experts can be useful.

Learning from an Expert strengthens knowledge and memory, and generates confidence in future difficult decisions.



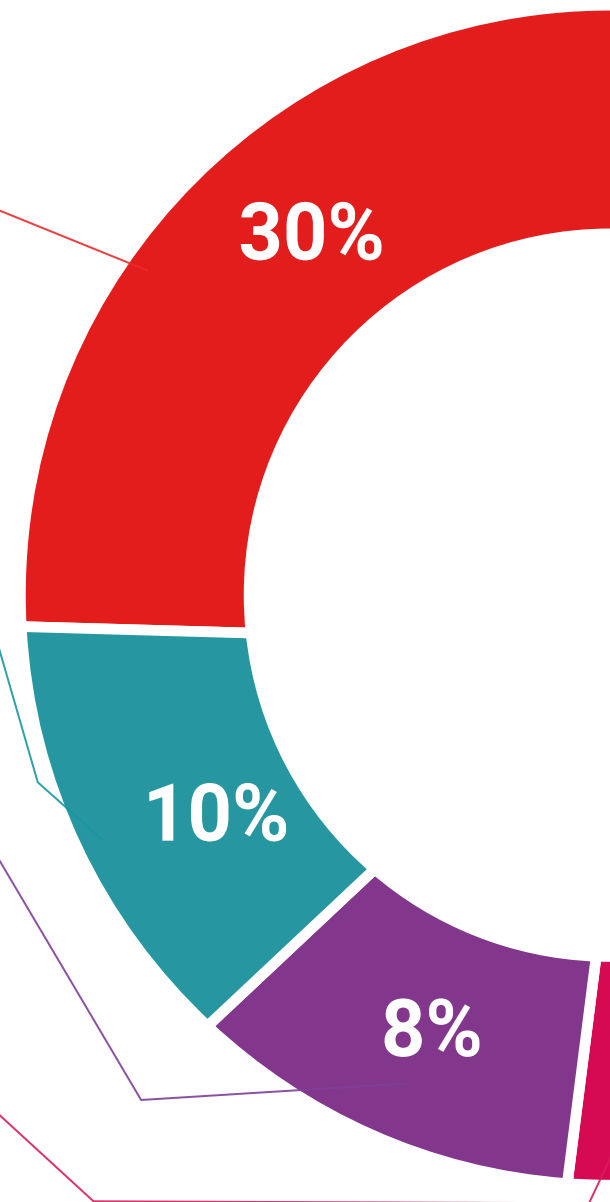
Practising Skills and Abilities

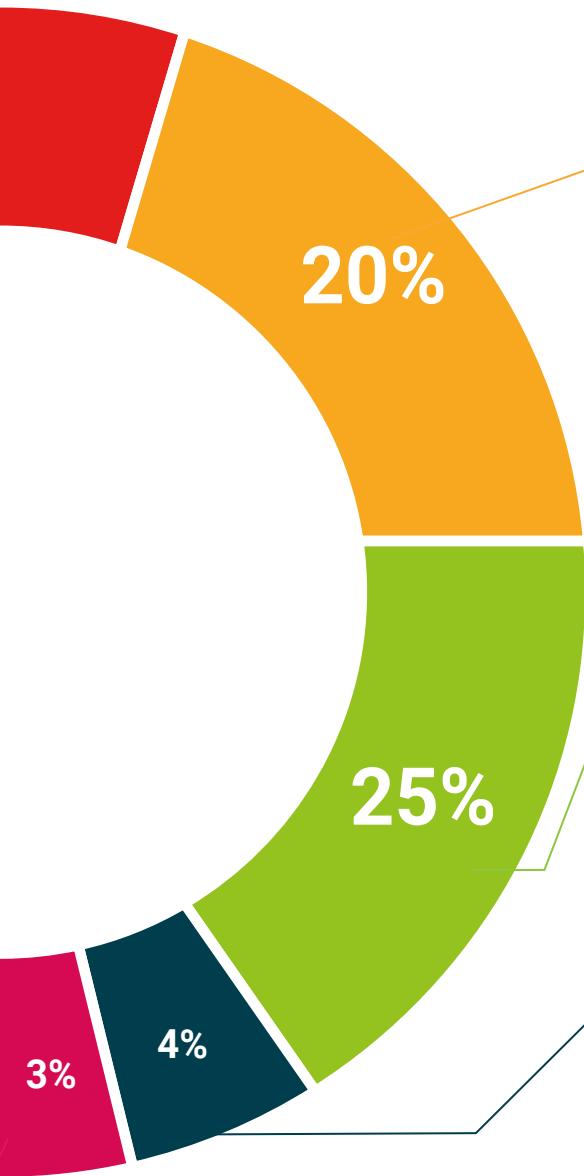
They will carry out activities to develop specific skills and abilities in each subject area. Exercises and activities to acquire and develop the skills and abilities that a specialist needs to develop in the context of the globalization that we are experiencing.



Additional Reading

Recent articles, consensus documents and international guidelines, among others. In TECH's virtual library, students will have access to everything they need to complete their course.





Case Studies

Students will complete a selection of the best case studies chosen specifically for this program. Cases that are presented, analyzed, and supervised by the best specialists in the world.



Interactive Summaries

The TECH team presents the contents attractively and dynamically in multimedia lessons that include audio, videos, images, diagrams, and concept maps in order to reinforce knowledge.

This exclusive educational system for presenting multimedia content was awarded by Microsoft as a "European Success Story".



Testing & Retesting

We periodically evaluate and re-evaluate students' knowledge throughout the program, through assessment and self-assessment activities and exercises, so that they can see how they are achieving their goals.



05 Certificate

The Postgraduate Diploma in Programming guarantees students, in addition to the most rigorous and up-to-date education, access to a Postgraduate Diploma issued by TECH Technological University.



“

Successfully complete this program and receive your university qualification without having to travel or fill out laborious paperwork”

This **Postgraduate Diploma in Programming** contains the most complete and up-to-date program on the market.

After the student has passed the assessments, they will receive their corresponding **Postgraduate Diploma** issued by **TECH Technological University** via tracked delivery*.

The diploma issued by **TECH Technological University** will reflect the qualification obtained in the Postgraduate Diploma, and meets the requirements commonly demanded by labor exchanges, competitive examinations, and professional career evaluation committees.

Title: **Postgraduate Diploma in Programming**

Official N° of Hours: **600 h.**



*Apostille Convention. In the event that the student wishes to have their paper diploma issued with an apostille, TECH EDUCATION will make the necessary arrangements to obtain it, at an additional cost.

future
health confidence people
education information tutors
guarantee accreditation teaching
institutions technology learning
community commitment
personalized service innovation
knowledge present quality
development languages
virtual classroom



Postgraduate Diploma Programming

- » Modality: online
- » Duration: 6 months
- » Certificate: TECH Technological University
- » Dedication: 16h/week
- » Schedule: at your own pace
- » Exams: online

Postgraduate Diploma Programming