Universidad Nacional San Cristobal de Huamanga (UNSCH)
Programa Profesional de
Ciencia de la Computación
Sílabo 2024-II

1. CURSO

CS113. Computer Science II (Mandatory)

2. INFORMACIÓN GENERAL

2.1 Curso	:	CS113. Computer Science II
2.2 Semestre	:	3^{rd} Semester.
2.3 Créditos	:	4
2.4 horas	:	2 HT; 4 HP;
2.5 Duración del periodo	:	16 semanas
2.6 Condición	:	Mandatory
2.7 Modalidad de aprendizaje	:	Face to face
2.8 Prerrequisitos	:	CS112. Computer Science I. (2^{nd} Sem)

CS112. Computer Science I. (2^{nd} Sem)

3. PROFESORES

Atención previa coordinación con el profesor

4. INTRODUCCIÓN AL CURSO

This is the third course in the sequence of introductory courses in computer science. This course is intended to cover Concepts indicated by the Computing Curriculum IEEE (c) -ACM 2001, under the functional-first approach. The object-oriented paradigm allows us to combat complexity by making models from abstractions of the problem elements and using techniques such as encapsulation, modularity, polymorphism and inheritance. The Dominion of these topics will enable participants to provide computational solutions to design problems simple of the real world.

5. OBJETIVOS

- Introduce the student in the fundaments of the paradigm of object orientation, allowing the assimilation of concepts necessary to develop an information system

6. RESULTADOS DEL ESTUDIANTE

1) Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions. (Usage)

3) Communicate effectively in a variety of professional contexts.. (Usage)

5) Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline. (Usage)

7. TEMAS

Unidad 1: Advanced STL (8)	
Resultados esperados: 1	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Associative Containers (std::set, std::map, std::unordered_set, std::unordered_map). • Adapters (std::stack, std::queue, std::priority_queue). • Advanced STL Algorithms. • Functors and Predicates.	• Understand the use of associative containers. [Usar] • Implement programs that use STL adapters. [Usar] • Apply advanced STL algorithms. [Usar] • Use functors and predicates with the STL. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

Unidad 2: Advanced Templates (7)	
Resultados esperados: 1	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Template Metaprogramming. • SFINAE (Substitution Failure Is Not An Error). • Perfect Forwarding.	• Apply template metaprogramming to solve complex problems. [Usar] • Understand and use SFINAE for template selection. [Usar] • Use Perfect Forwarding for efficient argument passing. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

Unidad 3: Variadic Templates (12)	
Resultados esperados: 1	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Introduction to Variadic Templates. • Variadic Template Functions. • Variadic Template Methods. • Variadic Template Classes. • Classes inheriting from variable lists of variadic templates. • Example: Implementing a custom tuple class.	• Understand the concept of variadic templates. [Familiarizarse] • Implement variadic functions. [Usar] • Design classes with variadic methods. [Usar] • Create classes with variadic templates. [Usar] • Implement inheritance with variable lists of variadic templates. [Usar] • Apply variadic templates to solve real-world problems. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

Unidad 4: Move Semantics and Rvalue References (5)	
Resultados esperados: 1	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Lvalues and Rvalues. • Rvalue References. • Move Semantics. • Move Constructors and Move Assignment Operators. • Perfect Forwarding (review).	• Explain move semantics and its purpose in C++. [Familiarizarse] • Define and use rvalue references. [Usar] • Analyze the performance implications of using move semantics. [Evaluar] • Implement move constructors and move assignment operators for custom classes. [Usar] • Apply move semantics to optimize resource management in C++ programs. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

Unidad 5: Design Patterns (Creational and Structural) (6)	
Resultados esperados: 1	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Singleton, Factory, Builder. • Adapter, Decorator, Facade.	• Understand and apply creational design patterns: Singleton, Factory, Builder. [Usar] • Understand and apply structural design patterns: Adapter, Decorator, Facade. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

Unidad 6: Functors (3)	
Resultados esperados: 1,3	
Temas	**Objetivos de Aprendizaje (*Learning Outcomes*)**
• Definition of Functors. • Functors and Templates. • Passing Functors to Functions using parameters. • Passing Functors to Functions using templates. • Passing Functors to Classes using parameters. • Passing Functors to Classes using templates. • Examples and Applications.	• Introduction to Functors. [Usar] • Using Functors as parameters to functions and classes. [Usar] • Using Functors in functions and classes through templates. [Usar]
Lecturas : [**Stroustrup2013**], [MJo19], [**Deitel17**]	

8. PLAN DE TRABAJO

8.1 Metodología

Se fomenta la participación individual y en equipo para exponer sus ideas, motivándolos con puntos adicionales en las diferentes etapas de la evaluación del curso.

8.2 Sesiones Teóricas

Las sesiones de teoría se llevan a cabo en clases magistrales donde se realizarán actividades que propicien un aprendizaje activo, con dinámicas que permitan a los estudiantes interiorizar los conceptos.

8.3 Sesiones Prácticas

Las sesiones prácticas se llevan en clase donde se desarrollan una serie de ejercicios y/o conceptos prácticos mediante planteamiento de problemas, la resolución de problemas, ejercicios puntuales y/o en contextos aplicativos.

9. SISTEMA DE EVALUACIÓN

********* EVALUATION MISSING ********

10. BIBLIOGRAFÍA BÁSICA

[MJo19] Nicolai M.Josuttis. *C++17-The Complete Guide*. 1st. 2019.