UNIVERSITY VISION

A trailblazer in arts, science and technology in the region.

UNIVERSITY MISSION

The University shall primarily provide advance instruction and professional training in science and technology, agriculture, fisheries, education and other related field of study. It shall undertake research and extension services, and provide progressive leadership in its area of specialization.

UNIVERSITY GOAL

To produce graduates with excellence and dignity in arts, science and technology.

UNIVERSITY OBJECTIVES

a. Enhance competency development, commitment, professionalism, unity and true spirit of service for public accountability, transparency and delivery of quality services;
b. Provide relevant programs and professional trainings that will respond to the development needs of the region;
c. Strengthen local and international collaborations and partnerships for borderless programs;
d. Develop a research culture among faculty and students;
e. Develop and promote environmentally-sound and market-driven knowledge and technologies at par with international standards;
f. Promote research-based information and technologies for sustainable development;
g. Enhance resource generation and mobilization to sustain financial viability of the university.

Program Objectives and its relationship to University Goals:

PROGRAM OBJECTIVES (PO)	OBJECTIVES						
A graduate of BS in Information System can:	a	B	c	d	e	f	G
a. Design and implement business information system;	/	/			/		/
b. Perform theoretical and practical skills in innovating latest technology in computing;	/	/	/	/	/	/	/
c. Design industry based services and technology that will promote advancement and development to the community;	/	/	/	/	/	/	/
d. Demonstrate the code of conduct as well as social and legal aspects of Information System.	/	/	/	/	/	/	/

1. Course Code   : CC 114
2. Course Title   : Data Structure and Algorithm
3. Prerequisite   : CC 113
4. Credits   : 3 UNITS

This course aims to equip students with the foundational principles and practical skills necessary to understand, design, and implement efficient data structures and algorithms in C++. The course aligns with the BS in Information Systems (BSIS) program objectives by preparing students to use computational thinking and programming techniques in business and industry settings, fostering their ability to create robust information systems solutions, and promoting ethical standards in technology us

6. Course Learning Outcomes and Relationships to Program Educational Objectives

Course Learning Outcomes	Program Objectives			
At the end of Semester the student can:	a	b	c	d
a. Apply fundamental and advanced data structures and algorithms in designing efficient information systems using C++.	/	/	/	/
b. Analyze and evaluate data structures and algorithmic choices to address specific needs in business information systems	/	/	/	/
c. Implement solutions that optimize data handling and processing in industry-aligned projects.	/	/	/	
d. Exhibit professionalism and adhere to ethical standards in computing and information systems development.	/	/	/	/

7. Course Content

Course Objectives, Topics, Time Allotment	Desired Student Learning Outcomes	Outcomes-Based Assessment (OBA) Activities	Evidence of Outcomes	Course Objectives	Program Outcomes	Values Integration
Topic: SKSU VMGO, Classroom Policies, Course Overview, Course Requirements, Grading System (2 hours)						
1.1 Discuss the VMGO of the university, classroom policies, scope of the course, course requirements and grading system	1.1 Student can be aware of and appreciate of the university's VMGO, classroom policies, course overview, requirements and grading system.	Individual participation in class discussion and group presentation	Group and individual discussions			Value of appreciation
2. Understanding Data Structures and Algorithm Concepts (10 hrs)						
2.1. Introduce students to the fundamental principles of data structures and algorithms, building the theoretical foundation necessary for efficient data organization and problem-solving

2.2. Gain a strong theoretical | 2.1. Student can identify and describe various data structures and their real-world applications in computing.

2.2. Student can explain the relationship between data structures and algorithms, including how they | Individual participation in class discussion and group presentation

Lectures
Interactive Sessions
Case Analysis | Group and individual discussions

Interactive Sessions
Case Analysis rubrics | a | d | Value of appreciation |

grounding in data structures, which is essential for developing innovative technology solutions in computing. 2.3. Discuss and emphasizes the role of data structures and algorithms in designing scalable and reliable business systems, an essential skill for information systems. 2.4. Explain the ethical implications of efficient and responsible data handling, which aligns with professional standards in Information Systems	support efficient problem-solving. 2.3. Student can analyze how data structures and algorithms are used in business information systems to optimize performance and reliability. 2.4. Student can demonstrate ethical and responsible perspectives on data organization and algorithmic efficiency.					
3. C++ Data Types (16 hrs)						
3.1. Equip students with practical skills	3.1. Student can use C++ primitive	Individual	Group and	a,b,d	a, d	Value of appreciation

in C++ by exploring primitive and derived data types, including pointers, structs, and classes, enabling them to manage data efficiently in program design. 3.2. Provide practical programming skills essential for implementing data management solutions in real-world information systems. 3.3. Discuss and understand memory management and data organization through C++ data types, students are prepared to design business information systems that are both effective and efficient. 3.4. Introduce	and derived data types to manage and organize information effectively. 3.2. Student can implement basic C++ programs that demonstrate an understanding of data types, memory allocation, and data management. 3.3. Student can apply pointers, structs, and classes in C++ to structure data in a way that optimizes storage and retrieval, supporting scalable system design. 3.4. Student demonstrate awareness of ethical practices in data handling and	participation in class discussion and group presentation Lectures Interactive Sessions Case Analysis	individual discussions Interactive Sessions Case Analysis rubrics			

				a,b,d	a, d	Unity and teamwork
common practices on memory usage, encouraging ethical programming practices in line with industry standards.	memory management, considering the impact on system efficiency and security.					
4. Data Structure Data Types (23 Hour)						
4.1. Introduce core data structures, such as arrays, linked lists, stacks, and queues, emphasizing their applications in building robust and responsive information systems. 4.2. Enhance students' problem-solving skills by teaching them how to select and implement appropriate data structures for specific business applications. 4.3. Gain knowledge about the creation of optimized business	4.1. Stundents can Differentiate between data structure types (e.g., arrays, linked lists, stacks, queues) and choose the most appropriate one for specific programming tasks in information systems. 4.2. Students can construct and implement these data structures in C++, with a focus on building efficient and responsive programs.	Individual participation in class discussion and group presentation Lectures Interactive Sessions Case Analysis	Group and individual discussions Interactive Sessions Case Analysis rubrics			Unity and teamwork Value of appreciation

| information systems, fostering innovation in industry-aligned technology solutions.

4.4. Encouraged to consider the scalability and ethical implications of data structure choices, which aligns with the responsible management of resources in professional settings.
. | 4.3. Students can evaluate the advantages and limitations of each data structure type, particularly in scenarios related to business information systems.

4.4. Students can reflect on ethical considerations in data structure selection, particularly in terms of resource management and system reliability | | | | | |

5.Topic: Algorithms (23 hours)

				a,b,d	a, d	
5.1. Provide a comprehensive understanding of algorithms, including expression evaluation and sorting, to optimize data processing and improve the performance of information systems. 5.2. Equips students with essential skills in algorithmic efficiency, preparing them to design systems that handle data quickly and effectively, which is critical in business contexts.	5.1. Student can apply PEMDAS rules to evaluate expressions in infix, prefix, and postfix notation, ensuring correct computational order. 5.2. Students can select and implement various sorting algorithms (Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort, and Heap Sort) in C++ for different data scenarios. 5.3. Student can analyze the time complexity of each algorithm to	Individual participation in class discussion and group presentation Lectures Situational Analysis Case Analysis Interactive Sessions	Group and individual discussions Interactive Sessions, Situational and Case Analysis rubrics			Value of participation, teamwork, Hard work, Resourcefulness, accomplishments and appreciation

5.3. Discuss about the focus on optimization aligns with industry needs for responsive systems, supporting the advancement of technology that meets organizational and community demands. 5.4. Gain insight into ethical programming practices by learning to write clean, efficient, and responsible code that aligns with industry standards.	determine the most efficient solution for specific business or data-centric applications. 5.4. Students can write efficient, maintainable code, focusing on algorithmic responsibility and ethical data processing practices in line with industry standards.				

6. Hashing and Basic Applications (16 hours)

6.1. Teach students the principles of hashing, hash tables, and collision resolution, emphasizing the role of hashing in fast data retrieval within information systems. 6.2. Discuss and implement the key hashing techniques, enabling students to build high-performance systems that support large-scale data management—essential for business applications. 6.3. Learn practical skills to implement hash tables, preparing them to contribute to	6.1. Students can explain the concept of hashing and how it supports rapid data retrieval in systems where large datasets are handled. 6.2. Students can design and implement basic hash tables using C++ to manage key-value pairs efficiently. 6.3. Students can apply different collision handling methods, such as chaining and open addressing, to maintain data integrity in hash tables. 6.4. Students can evaluate ethical considerations in data security and	Individual participation in class discussion and hands on Lectures Situational Analysis Case Analysis Group Demonstration Interactive Sessions	Group and individual discussions Interactive Sessions ,Situational Analysis, Case Analysis, and demonstration rubrics	a,b,d	b, d	Value of participation, Hard work, Resourcefulness, accomplishments, teamwork and communication

technology that meets industry and community needs through efficient data handling. 6.4. Reinforce ethical programming practices, as students consider data privacy and integrity in hashing, aligned with legal and social standards in Information Systems.	integrity when using hashing techniques, aligning with legal and social standards in Information Systems.					
Total Hours	**Lecture: 36** **Laboratory: 54** **Exam: 4** **Total : 94 hours**					

Course Evaluation

Course Requirements:

Grading System:

MIDTERM		**FINAL TERM**	
Exam	-50%	Exam	-50%
Class Work	-30%	Class Work	- 30%
Attendance	- 10%	Attendance	- 10%
Quizzes	-10%	Quizzes	-10%

MTG+FTG/2=FG
Schedule of Examination

Midterm   -
Final Term   -
Classes End   -

References:

Online Resources

GeeksforGeeks. (n.d.). *Data Structures and Algorithms*. Retrieved from https://www.geeksforgeeks.org/
TutorialsPoint. (n.d.). *Data Structures Tutorial*. Retrieved from
https://www.tutorialspoint.com/data_structures_algorithms/
Programiz. (n.d.). *Learn Data Structures and Algorithms*. Retrieved from https://www.programiz.com/dsa

Textbook:

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms*. MIT Press.
Sedgewick, R., & Wayne, K. (2011). *Algorithms*. Addison-Wesley.
Knuth, D. E. (1997). *The Art of Computer Programming*. Addison-Wesley.
Stroustrup, B. (2013). *The C++ Programming Language*. Addison-Wesley.
Lippman, S. B., Lajoie, J., & Moo, B. E. (2012). *C++ Primer*. Addison-Wesley.
Josuttis, N. M. (2012). *The C++ Standard Library: A Tutorial and Reference*. Addison-Wesley.
Weiss, M. A. (2013). *Data Structures and Algorithm Analysis in C++*. Pearson.
Horowitz, E., Sahni, S., & Anderson-Freed, S. (2006). *Fundamentals of Data Structures in C*. Silicon Press.

Supplemental:

LaFore, R. (2018). *Data Structures and Algorithms in Java*. Sams Publishing.
Goodrich, M. T., Tamassia, R., & Mount, D. M. (2011). *Data Structures and Algorithms in C++*. Wiley.

Prepared and Reviewed by:

ALEXIS D. APRESTO, PhD
Faculty / Program Head, BSIS

Recommending Approval:

ELBREN O. ANTONIO, DIT   **ROMMEL M. LAGUMEN, PhD**
Dean, College of Computer Studies   Director, SKSU-Isulan Campus