PROBLEM SOLVING TECHNIQUES USING C

---

1. Knowledge: Students will be able to describe the generations of programming languages, including machine language, assembly language, procedural language, and object-oriented language.
2. Comprehension: Students will be able to explain the features and structure of a C program, including character set, identifiers, keywords, variables, constants, and the C library.
3. Application: Students will be able to demonstrate the use of different data types in C, including primitive, user-defined, enumerated, and type casting.
4. Analysis: Students will be able to analyze different types of operators in C, including their precedence, associativity, and their use in expressions.
5. Synthesis: Students will be able to create programs that involve input-output operations using standard functions, escape sequences, and format specifiers.
6. Evaluation: Students will be able to evaluate and select appropriate decision-making statements such as if-else, switch-case, and the ternary operator in C programs.
7. Analysis: Students will be able to analyze and implement iterative statements such as for, while, and do-while loops, along with control statements like break and continue.
8. Synthesis: Students will be able to develop functions in C, including their declaration, calling, and definition, with an understanding of recursive functions.
9. Application: Students will be able to apply concepts of arrays and pointers in C programming to manipulate data efficiently.
10. Analysis: Students will be able to analyze different storage classes in C and their applications in programming.
11. Evaluation: Students will be able to evaluate and implement structures and unions in C to organize and store data effectively.
12. Create: Students will be able to create and use macros in C to define reusable code snippets for efficient programming.