



PROBLEM SOLVING WITH C UE23CS151B

Prof. Sindhu R Pai

Department of Computer Science and Engineering

PROBLEM SOLVING WITH C

Functions in C

Prof. Sindhu R Pai

Department of Computer Science and Engineering

PROBLEM SOLVING WITH C

Functions



1. Introduction to functions
2. Types of Functions
3. Function Definition, Call and Declaration

PROBLEM SOLVING WITH C

Functions



Introduction

- A sub-program to carry out a specific task
- Functions break large computing tasks into smaller ones.
- Enable people to build on what others have done instead of starting from scratch
- **Benefits:**
 - Reduced Coding Time – Code Reusability
 - Divide and Conquer - Manageable Program development
 - Reduced Debugging time
 - Treated as a black box - The inner details of operation are invisible to rest of the program

PROBLEM SOLVING WITH C

Functions



Types of Functions

- **Standard Library Functions**

Must Include appropriate header files to use these functions.
Already declared and defined in C libraries
printf, scanf, etc..

- **User Defined functions**

Defined by the developer at the time of writing program
Developer can make changes in the implementation

PROBLEM SOLVING WITH C

Functions



Function Definition, Call and Declaration

Function Definition

- Provides actual body of the function
- Function definition format

```
return-type function-name( parameter-list )
{
    declarations and statements
}
```
- Variables can be declared inside blocks
- Coding examples

Function Definition, Call and Declaration

Function Call

- Function can be called by using function name followed by list of arguments (if any) enclosed in parentheses
- Function call format
function-name(list of arguments);
- The arguments must match the parameters in the function definition in its type, order and number.
- Multiple arguments must be separated by comma. Arguments can be any expression in C
- Coding examples

PROBLEM SOLVING WITH C

Functions



Function Definition, Call and Declaration

Function Declaration/ Prototype

- All functions must be declared before they are invoked or called.
- Function can be declared by using function name followed by list of parameters (if any) enclosed in parentheses
- Function declaration format
return_type function_name (parameters list);
- Use of identifiers in the declaration is optional.
- The parameter names do not need to be the same in declaration and the function definition.
- The types must match the type of parameters in the function definition in number and order.
- Coding examples



THANK YOU

Prof. Sindhu R Pai

Department of Computer Science and Engineering

sindhurpai@pes.edu