**Constants and Macros Assignment**

1. **Write a function macro to find the smallest number in an array of integers.**

**A computer screen shot of a program code

Description automatically generated**

**A black background with white letters

Description automatically generated**

**2. What are the differences between macros and constant. Can you replace a constant with a macro and vice versa? Give examples for your statements**

Macros:

· Macros are preprocessor directives, expanded by the preprocessor before compilation.

· They are typically used to define simple constants or functions.

· Macros are not type-safe and can cause issues if not used carefully, such as operator precedence problems.

Constants:

· Constants are fixed values defined by the const keyword and are type-safe.

· Constants are evaluated during runtime, and their value cannot be changed once assigned.

Can you replace a constant with a macro and vice versa?

· You can replace constants with macros for simple values. However, replacing complex constants (like float values or array sizes) with macros may lead to unintended behavior.

· Replacing macros with constants may be possible, but macros can handle complex expressions, whereas constants are limited to fixed values.

Example:

// Constant Example

const int x = 10; // Type-safe, can be used in any expression.

// Macro Example

#define X 10 // This is just a substitution; no type check.

**3. Refer macro below**

**#define MYPROD(x) (x \*x)**

**WAP to invoke the above macro with inputs as below and display the result.**

**a. MYPROD(2+1)**

**b. MYPROD(6+1)**

**Do you get the expected answers as 9 and 49 in case a. and case b.?**

**If not modify the code to produce the expected results. in above case.**

A screenshot of a computer program

Description automatically generated

A black screen with white text

Description automatically generated

**4. Write macro definitions with arguments for calculation of area of a triangle and circle.**

**a. Use macros for both constants as well as formula evaluations.**

**b. Store these macro definitions in a header file and invoke the macros from the main function.**

**A screen shot of a computer

Description automatically generated**

**5. Define a macro name MYPRINT as below.**

**#define MYPRINT(x) printf(x)**

**Use the above macro conditionally only if a macro CUST\_PRINT is defined , otherwise not to be used.**

**For eg refer the code and comments below.**

**int main()**

**{**

**MYPRINT("Hello World"); // will be displayed only when CUST\_PRINT is defined**

**printf("Test"); // will be displayed always irrepective of CUST\_PRINT**

**return 0;**

**}**

**Add the code to demonstrate the above behaviour.**

**A screen shot of a computer screen

Description automatically generated**

**A black screen with white text

Description automatically generated**

**6. Identify and use the macros to display**

**a. file name**

**b. function name**

**c. line of code**

**Show the usage with a code example**

**A computer screen shot of a program

Description automatically generated**

**A screen shot of a computer

Description automatically generated**