**Assigment:1 DATE: 02/01/2024**

**AIM:Add comments to program.**

**Pratical :1**

**#include <stdio.h>**

*// Swap Function Definition*

**void swap(int \*a, int \*b) {**

**int temp = \*a;**

**\*a = \*b;**

**\*b = temp;**

**}**

*// Main Function*

**int main() {**

**int x = 5, y = 10;**

*// Print values before swap*

**printf("Before Swap: x = %d, y = %d\n", x, y);**

*// Call Swap Function*

**swap(&x, &y);**

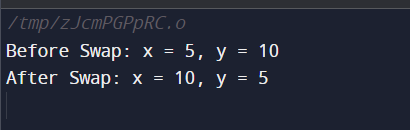
*// Print values after swap*

**printf("After Swap: x = %d, y = %d\n", x, y);**

**return 0;**

**}**

**OUTPUT:**



**Pratical :2**

**#include <stdio.h>**

*// main function*

**int main() {**

*// Initialize an integer array*

**int arr[] = {1, 2, 3, 4, 5};**

*// Declare and initialize an integer pointer to the first element of the array*

**int \*ptr = arr;**

*// Initialize a variable to store the sum of array elements*

**int sum = 0;**

*// Iterate through the array using the pointer and calculate the sum*

**for (int i = 0; i < 5; ++i) {**

**sum += \*ptr;** *// Accumulate the value pointed to by ptr*

**ptr++;** *// Move the pointer to the next element*

}

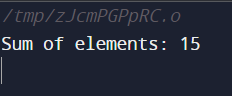
*// Print the sum of array elements*

**printf("Sum of elements: %d\n", sum);**

**return 0;**

**}**

**OUTPUT:**



**Pratical :3**

*// Header file*

**#include <stdio.h>**

*// Main function*

**int main() {**

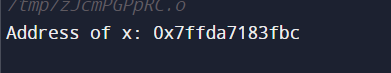
**int x = 10;**

**printf("Address of x: %p\n", (void\*)&x);** *// Prints the memory address of variable x*

**return 0;**

**}**

**OUTPUT:**



**Pratical :4**

**#include <stdio.h>**

*// Function to calculate the length of a string*

**int stringLength(char \*str) {**

**int length = 0;** *// Initialize length to zero*

**while (\*str != '\0') {**

**length++;** *// Increment length for each character*

**str++;**  *// Move to the next character in the string*

**}**

**return length;** *// Return the final length*

**}**

*// Main function*

**int main() {**

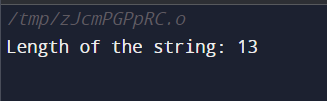
**char str[] = "Hello, World!";** *// Declare and initialize a string*

**printf("Length of the string: %d\n", stringLength(str));** *// Print the length of the string*

**return 0;** *// Indicate successful program execution*

**}**

**OUTPUT:**



**Pratical :5**

**#include <stdio.h>**

**#include <stdlib.h>**

*// Dynamic memory allocation example*

**int main() {**

*// Declare a pointer to int*

**int \*ptr;**

*// Allocate memory for an integer*

**ptr = (int \*)malloc(sizeof(int));** *// Allocating memory of size int*

*// Check if memory allocation is successful*

**if (ptr == NULL) {**

**printf("Memory allocation failed.\n");** *// Error message if allocation fails*

**return 1;**

**}**

*// Store a value in the dynamically allocated memory*

**\*ptr = 10;**

*// Print the value stored in the allocated memory*

**printf("Value stored at dynamically allocated memory: %d\n", \*ptr);**

*// Free the allocated memory to avoid memory leaks*

**free(ptr);**

**return 0;**

**}**

**OUTPUT:**

