Programming Fundamentals Lab Exam **Marks: 100**

Name: _____ **Roll No: _____**

Professor: Imran Javed ***Marks Obtained: _____***

1. What is an array, and how is it different from a normal variable in C? (5)

An array is a collection of elements of the same data type stored at contiguous memory locations. Unlike normal variables, which can store only one value at a time, arrays can store multiple values.

2. What will happen if you try to access an array element beyond its size? (5)

Accessing an array element beyond its size leads to undefined behavior, which may result in accessing garbage values or causing a segmentation fault.

3. How can you pass an array to a function in C? (5)

You can pass an array to a function by passing the pointer to its first element. For example, void function(int arr[], int size).

4. What is pointer arithmetic, and how does it work? (10)

Pointer arithmetic involves operations on pointers, such as addition and subtraction. For example, adding 1 to a pointer increments the pointer to point to the next memory location of the type it points to.

5. What is the difference between int *ptr and int **ptr? (5)

int *ptr is a pointer to an integer, while int **ptr is a pointer to a pointer to an integer (a pointer to a pointer).

6. How do you dynamically allocate memory for an integer array of size 10 in C? (10)

You can dynamically allocate memory for an integer array of size 10 using int *arr = (int *) malloc(10 * sizeof(int));.

7. What is the purpose of the typedef keyword when used with structures? (5)

The typedef keyword is used to create an alias for a data type, including structures, making the code more readable and easier to use. For example:

```
typedef struct {
   int id;
   char name[50];
   float marks;
} Student;
```

8. What is the difference between a while loop and a do-while loop? (10)

A while loop checks the condition before executing the loop body, whereas a do-while loop executes the loop body at least once before checking the condition.

9. How can you exit from a loop prematurely in C? (5)

You can exit from a loop prematurely using the break statement.

10. What is the base case in recursion? (5)

The base case is the condition that stops the recursion, preventing an infinite loop.

11. How can you prevent infinite recursion? (5)

you can prevent infinite recursion by ensuring that there is a base case that will eventually be met.

12. What is the difference between direct and indirect recursion? (10)

Direct recursion occurs when a function calls itself directly, while indirect recursion occurs when a function calls another function that eventually calls the original function.

13. write a program to swap two variables without using third variable (20)

You can swap two variables without using a third variable by leveraging arithmetic operations or bitwise XOR operations.

Example:

```
#include <stdio.h>

int main() {
   int a, b;
   printf("Enter two numbers to swap: ");
   scanf("%d %d", &a, &b);
   printf("Before swapping: a = %d, b = %d\n", a, b);
   // Swapping using arithmetic operations
   a = a + b;
   b = a - b;
   a = a - b;
   printf("After swapping: a = %d, b = %d\n", a, b);
   return 0;
}
```