Name: KRISHANU DEY
Reg. No: 2021CA055

Date
Page 1

P and PS (Theory)

Assignment No. 2

1) Is it possible to have a function as a parameter in another function?

Ans- No, we cannot pass the function as an argument to another function. But we can pass the reference of a function as a parameter by using a function pointer. This process is known as call by reference as the function parameter is passed as a pointer that holds the address of arguments.

Syntax: (type) (*pointer-name)(parameter);

2) What is the difference between Call by Value and Call by Reference?

Ans-

Call by Value	Call by Reference
• While calling a function, we pass values of variables to it.	• While calling a function, we pass address of variables to it.
• Value of variable in calling function is copied to dummy variable in called function	• Address of variable in calling function is copied to dummy variable of called function.
• Changes made on dummy variable have no effect on actual variable.	• Changes made on dummy variable also reflects on actual variables.

3) What are global variables and how do you declare them

Ans- Global variables are those variables which are declared outside of all the functions or block and can be accessed globally in a program. It can be accessed by any ~~program~~ function present in the program.

eg:
```
#include <stdio.h>
int a=50, b=40; // global variable declaration
void main() {
    printf("a=%d and b=%d" a, b);
}
```

4) Is that possible to add pointers to each other?

Ans- No, there is no possibility to add pointers to each other since, pointer contains address details there is no way to retrieve the value from this operation.

5) What are the ways to a null pointer that can be used in the C Programming language?

Ans- Some of the most common use cases for NULL are :-

a) To initialize a pointer variable when that pointer variable isn't assigned any valid memory address yet.
b) To check for a null pointer before accessing any pointer variable. By this we can perform error handling.
c) Also passed as an argument to a function.

6) Please explain the concept of Dangling Pointer in C? In how many ways can a pointer act as a Dangling Pointer?

Ans- A pointer pointing to a memory location that has been deleted (or freed) is called dangling pointer. There are three different ways where Pointer acts as dangling pointer:

a) De-allocation of memory.
b) Function call
c) Variable goes out of scope.

7) Compare arrays with pointers in the C Programming language.

Ans-

Array	Pointer
• type var_name [size];	• type * var_name;
• Collection of elements of similar data type.	• Store the address of another variable.
• It can decide the numbers of element it can store.	• It can store the address of only one variable.
• Arrays are static in nature	• Pointers are dynamic in nature
• An array of pointers can be generated.	• A pointer to an array can be generated.
• Memory allocation is in sequence.	• Memory allocation is random.

8) Explain Recursion in C.

Ans- Recursion is the process which comes into existence when a function calls a copy of itself to work on a smaller problem. Any function which calls itself is called recursive function, and such function calls are called recursive calls.

eg:-

```
#include <stdio.h>
int fact(int);          // recursive function declaration
int main() {
    int n, f;
    printf("Enter a number: ");
    scanf("%d", &n);
    f = fact(n);        // stores back result
    printf("factorial = %d", f);
}

// Recursive function definition..
int fact(int n)
{
  if (n==0)             // base case
    return 0;
  if (n==1)             // base case
    return 1;
  else
      return n * fact(n-1); // recursive call
}
```

—— X ——