**Introduction to Pointers Assignments**

**Mandatory**

**1. Refer the code snippet below. int main()**

**{**

**char arr=”hello hi “;**

**int \*ptr = arr;**

**printf(“sizeof ptr:%d, arr:%d”, sizeof(ptr), sizeof(arr));**

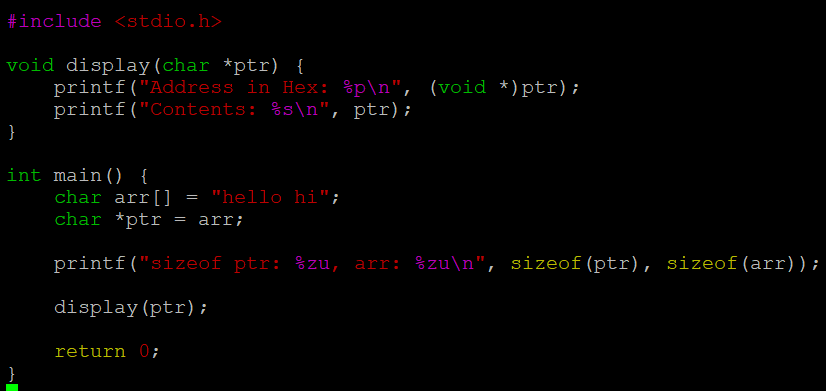
**display(ptr); // display the address in hex and contents using pointer**

**}**

**Perform the following.**

**a. Implement the display() function (Use the “0x%x” formatting specifier to print addresses in hexadecimal.)**

**b. comment on the sizeof(ptr) and sizeof(arr)**

****

**A screen shot of a computer code

Description automatically generated**

**2. Refer the code snippet below. int main()**

**#define MAX 100**

**#define SUCCESS 0**

**#define FAILURE 1**

**int main()**

**{**

**char arr[MAX] = “Learning C“;**

**char\*ptr = arr;**

**char appendstr[3]= “in my org”;**

**printf(“Address of ptr:%x”, ptr);**

**int ret = append(ptr, appendstr);// append the string**

**printf(“Address of ptr:%x”, ptr);**

**if (ret == SUCCESS)**

**{**

**display(ptr); // display the address in hex and contents using pointer**

**}**

**}**

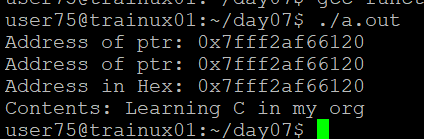
**Perform the following.**

**a. Implement the append() function to append the contents of the appendstr[] to arr using pointer.**

**[Note: append() should only use its content and not manipulate it. Contents should be retained even after the call]**

**A screen shot of a computer program

Description automatically generated**

****

**3. Refer the code in “pointer\_prg.c”. The functions swap\_nums() and swap\_pointers() are expected to swap the numbers and pointers respectively. But swap\_pointers() is currently not giving the expected results. Analyse and the fix the issue.**

To swap the address we need to pass by reference for that we need to take a double pointer to catch the address, then we get the required result

A computer screen with text on it

Description automatically generated

**A computer screen with text on it

Description automatically generated**