1. Write a function char \*mystrstr( char \*str1, char \*str2), which finds the first occurrence of the substring **str2** in the string **str1**. The terminating '\0' characters are not compared. **(10)**

## **Parameters:**

* **str1**-- This is the main C string to be scanned.
* **str2** -- This is the small string to be searched with-in **str1** string.

## **Return Value:**

This function returns a pointer to the first occurrence in **str1** of any of the entire sequence of characters specified in **str2**, or a null pointer if the sequence is not present in **str1**.

**Restriction:** You have to use Pointer syntax for implementing mystrstr function. You cannot declare any temporary char array in this function. You cannot use any function declared in <string.h>.

Now, use the following main function to test your implemented function.

|  |
| --- |
| int main() {  char \*str1;  char \*str2;  str1 = (char \*) malloc (100 \* sizeof(char));  str2 = (char \*) malloc (100 \* sizeof(char));    gets(str1);  gets(str2);   printf("%s\n", mystrstr(str1, str2));  free(str1);  free(str2);  return 0; } |

**Sample IO:**

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| CSE 102 CSE B Section  EEE C | CSE B Section |
| AAABCD  AABC | AABCD |
| ABBCCDD  BBCCAA | NULL |
| ABBBCDDD  DDD | DDD |