Assignment 1

Pragna Mamidipaka - EE20BTECH11026

Download all latex-tikz codes from

https://github.com/Pymamid/C−and−Data−Structures/blob/main/Assignment1/Assignment1.tex

1 Problem

(Q 29) Consider the following C function.

```
#include<stdio.h>
void fun1(char *s1, char *s2){
    char *tmp;
    tmp = s1;
    s1 = s2;
    s2 = tmp;
}
void fun2(char **s1, char **s2){
    char *tmp;
    tmp = *s1;
    *s1 = *s2;
    *s2 = tmp;
}
int main(){
    char *str1 = "Hi", *str2 = "Bye";
    fun1(str1, str2);
    printf("%s %s ", str1, str2);
    fun2(&str1, &str2);
    printf("%s %s", str1, str2);
    return 0;
}
```

The output of the program above is

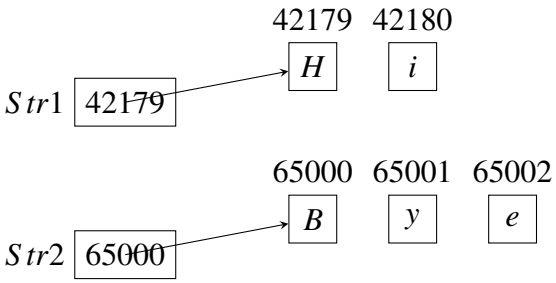
1) Hi Bye Bye Hi
2) Hi Bye Hi Bye
3) Bye Hi Hi Bye
4) Bye Hi Bye Hi

2 Solution

Answer : The output of the above program is:
1) Hi Bye Bye Hi

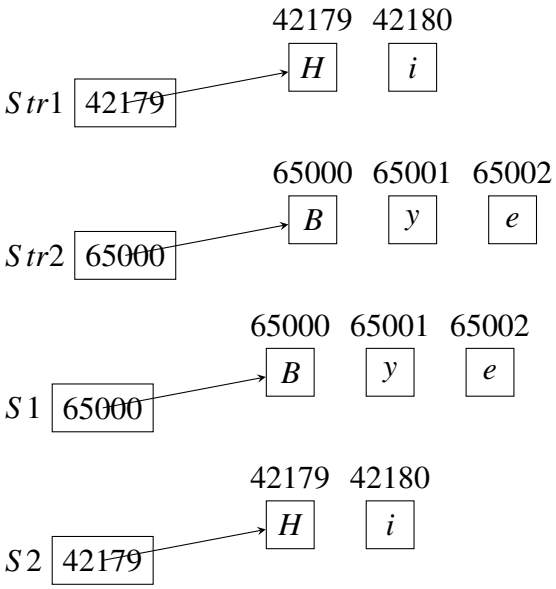
Explanation

Str1 and *Str2* before call to *fun1*:



The function *fun1* is call-by-value. In this mechanism, the values of actual parameters get copied to formal parameters and the modifications performed on formal parameters will not be updated on actual parameters. Therefore, *str1* and *str2* still point to their old values. *s1* and *s2* point to Bye and Hi respectively.

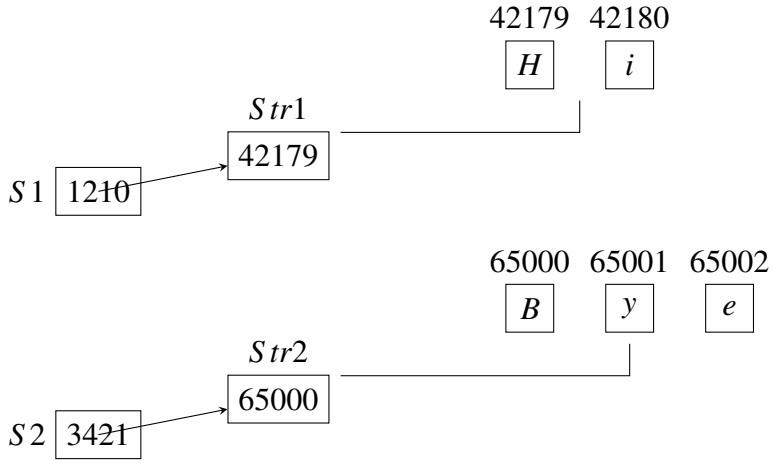
i.e., after call to *fun1*,
str1 points to **Hi**
str2 points to **Bye**



The function *fun2* is call-by-reference. In this mechanism, the address of actual parameters get copied to formal parameters. Therefore, the modifications performed via formal parameters will be updated on actual parameters. Therefore, the values in *str1* and *str2* get interchanged by calling the second function.

i.e., after call to *fun2*,
str1 points to **Bye**
str2 points to **Hi**

Address of str1 and str2 are passed to *fun2*:



fun2 swaps the values dereferenced by S1 and S2: