

Experiment 6-S

```
#include <stdio.h>
#include <math.h>
```

```
void REVERSE (char str[]) {
```

```
    int i, len;
```

```
    char temp;
```

```
    len = strlen(str);
```

```
    for (i=0; i<len/2; i++) {
```

```
        temp = str[i];
```

```
        str[i] = str[len - i - 1];
```

```
        str[len - i - 1] = temp;
```

}
{

```
int main() {
```

```
    char str[100];
```

Prinip ("-- String Reversal Program --- /n");
 Print ("Enter a string : ");
 f gets (str, size of (str), stdin);
 str [size of (str, " /n")] = 0

Teacher's Signature _____

Remarks:

~~REVERSE~~
REVERSE (str);

```
printf ("\\n Reversed String : '%s'", str);  
return 0;
```

1

Sept. 20, 1968.

~~in today's lesson~~

~~Step 10 - 100% 100%~~

Will focus on the following:

1935-61 - *1935-61*

Remarks:

Teacher's Signature _____

Online Compiler

The screenshot shows an online C compiler interface. The code editor on the left contains a C program named 'main.c' which implements a string reversal function. The output window on the right shows the execution results, including the reversed string 'olleH olleH'.

```
main.c
2 #include <string.h>
3
4 void REVERSE(char str[]) {
5     int i, len;
6     char temp;
7     len = strlen(str);
8     for (i = 0; i < len / 2; i++) {
9         temp = str[i];
10        str[i] = str[len - i - 1];
11        str[len - i - 1] = temp;
12    }
13 }
14
15 int main() {
16     char str[100];
17     printf("----String Reversal Program----\n");
18     printf("Enter a String: ");
19     fgets(str, sizeof(str), stdin);
20     str[strcspn(str, "\n")] = 0;
21
22     REVERSE(str);
23
24     printf("\nReversed String: %s", str);
25
26     return 0;
27 }
```

Output

```
----String Reversal Program----
Enter a String: Hello World

Reversed String: olleH olleH

--- Code Execution Successful ---
```