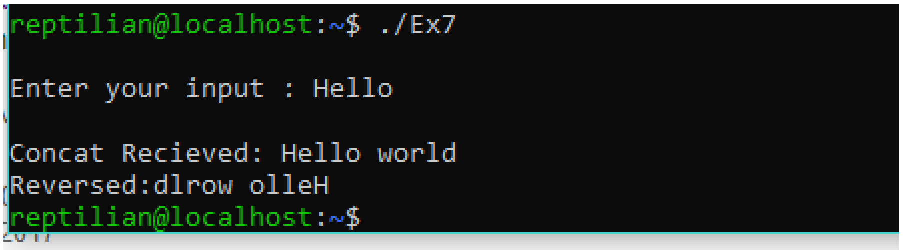
Exercise 7 Screenshot:



```
reptilian@localhost:~$ ./Ex7

Enter your input : Hello

Concat Recieved: Hello world
Reversed:dlrow olleH
reptilian@localhost:~$
```

Source code:

```
#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<string.h>

#include<sys/wait.h>

void reverse(char [], int, int);


int main()

{

        // We use 4 pipes

        // First pipe to send input string from parent

        // Second pipe to send concatenated string from child

        // Third pipe to send concat string from parent

        // Fourth pipe to send reversed string from child

        char fixed_str[] = " world";

        int pipe1[2], pipe2[2], pipe3[2], pipe4[2];

        pipe(pipe1);

        pipe(pipe2);

        pid_t p;

        p = fork();

        if (p < 0)
```

```
    {
        fprintf(stderr, "fork Failed");
        return 1;
    }
    // Parent process
    else if (p > 0)
    {
        pid_t p2;
        siginfo_t test;
        char concat2[107];
        char* input;
        printf("\nEnter your input : ");
        scanf ("%m[^\n]%*c", &input);
        close(pipe1[0]);
     write(pipe1[1], input, 100);
        close(pipe1[1]);
        waitid(P_PID,p,&test,WEXITED);
        close(pipe2[1]);
        read(pipe2[0], concat2, 107);
        close(pipe2[0]);
        pipe(pipe3);
        pipe(pipe4);
        p2 = fork();
        if (p2 < 0)
        {
            fprintf(stderr, "fork Failed");
            return 1;
        }
        else if (p2 > 0)
        {
              close(pipe3[0]);
              write(pipe3[1],concat2,107);
              close(pipe3[1]);
              waitid(P_PID,p2,&test,WEXITED);
              char output[107];
              close(pipe4[1]);
              if(read(pipe4[0], output, 107) == 0)
              {
              printf("\nError please try again");
              }
              close(pipe4[0]);
              printf("\nReversed:%s \n", output);
              free(input);
    }
    else // child process
    {
              char concat3[107];
              close(pipe3[1]);
read(pipe3[0], concat3, 107);
              close(pipe3[0]);
              printf("\nConcat Recieved: %s",concat3);
              int j = 0;
while(concat3[j] != '\0' && j < 106)
{
      j++;
}
              char *temp;
strcpy(temp, concat3);
reverse(temp,0, j-1);
              close(pipe4[0]);
```

```
                    write(pipe4[1], temp, 107);
                                    close(pipe4[1]);
                    exit(0);
                        }
            }
            // first child process
            else
            {
                        char input[100];
                        char concat[107];
                        close(pipe1[1]);
                        read(pipe1[0], input, 100);
                        close(pipe1[0]);
                        strcpy(concat, input);
                        strcat(concat, fixed_str);
                        close(pipe2[0]);
                        write(pipe2[1], concat, 107);
                        close(pipe2[1]);
                        exit(0);
            }
exit(0);
}

void reverse(char str1[], int index, int size)
{
    char temp;
    temp = str1[index];
    str1[index] = str1[size - index];
    str1[size - index] = temp;
    if (index == size / 2)
    {
        return;
    }
    reverse(str1, index + 1, size);
}
```