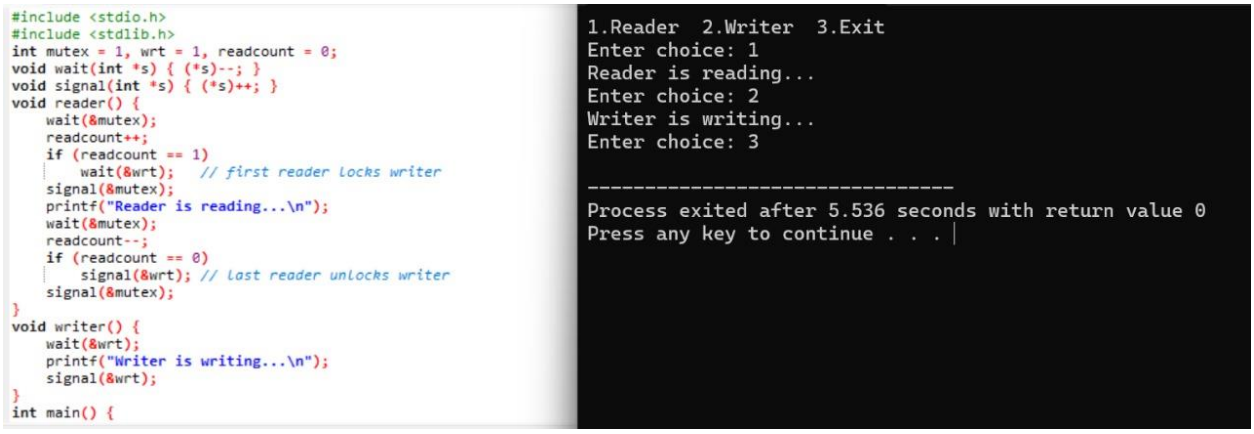
20. Construct a C program to simulate Reader-Writer problem using Semaphores.

```
#include <stdio.h>
#include <stdlib.h>
int mutex = 1, wrt = 1, readcount = 0;
void wait(int *s) { (*s)--; }
void signal(int *s) { (*s)++; }
void reader() {
   wait(&mutex);
   readcount++;
   if (readcount == 1)
      wait(&wrt);   // first reader locks writer
   signal(&mutex);
   printf("Reader is reading...\n");
   wait(&mutex);
   readcount--;
   if (readcount == 0)
      signal(&wrt); // last reader unlocks writer
   signal(&mutex);
}
void writer() {
   wait(&wrt);
   printf("Writer is writing...\n");
   signal(&wrt);
}
int main() {
   int ch;
```

```
    printf("1.Reader  2.Writer  3.Exit\n");
    while (1) {
        printf("Enter choice: ");
        scanf("%d", &ch);
        switch (ch) {
            case 1: if (wrt == 1 || readcount >= 0) reader();
                    else printf("Writer active!\n"); break;
            case 2: if (wrt == 1 && readcount == 0) writer();
                    else printf("Readers active!\n"); break;
            case 3: exit(0);
            default: printf("Invalid choice!\n");
        }
    }
}
```

OUTPUT-



```
#include <stdio.h>
#include <stdlib.h>
int mutex = 1, wrt = 1, readcount = 0;
void wait(int *s) { (*s)--; }
void signal(int *s) { (*s)++; }
void reader() {
    wait(&mutex);
    readcount++;
    if (readcount == 1)
        wait(&wrt);   // first reader locks writer
    signal(&mutex);
    printf("Reader is reading...\n");
    wait(&mutex);
    readcount--;
    if (readcount == 0)
        signal(&wrt); // last reader unlocks writer
    signal(&mutex);
}
void writer() {
    wait(&wrt);
    printf("Writer is writing...\n");
    signal(&wrt);
}
int main() {
```

```
1.Reader  2.Writer  3.Exit
Enter choice: 1
Reader is reading...
Enter choice: 2
Writer is writing...
Enter choice: 3

--------------------------------
Process exited after 5.536 seconds with return value 0
Press any key to continue . . .
```