

src\main.c

```
1
2 // task 2, EMB1 assignment 10
3
4 #include <stdio.h>
5 #include <avr/io.h>
6 #include <util/delay.h>
7 #include <stdlib.h>
8 #include "usart.h"
9
10 typedef struct { // define structure
11     int Re;
12     int Im;
13 } complex_t;
14
15 void read_complex(complex_t *com_number)
16 {
17     printf("Type in real part of number:");
18     scanf("%d",&(com_number->Re));
19     printf("Type in imaginary part of number:");
20     scanf("%d",&(com_number->Im));
21 }
22
23 complex_t add_complex(complex_t n1, complex_t n2)
24 {
25     complex_t result;
26     result.Re=n1.Re+n2.Re;
27     result.Im=n1.Im+n2.Im;
28     return result;
29 }
30
31 complex_t mul_complex(complex_t n1, complex_t n2)
32 {
33     complex_t result;
34     result.Re=n1.Re*n2.Re-n1.Im*n2.Im;
35     result.Im=n1.Re*n2.Im+n2.Re*n1.Im;
36     return result;
37 }
38
39
40 int main(void) {
41
42     unsigned char i; // initiate variables
43
44     uart_init();
45     io_redirect();
46
47     complex_t ar[2],result;
48 }
```

```
49 while(1) // start program loop
50 {
51     for(i=0;i<2;i++) //loop for reading complex number, first real, then imaginary
52     {
53         printf("Number %hu\n", i+1);
54         read_complex(&ar[i]);
55     }
56     for(i=0;i<2;i++) //loop for printing complex number
57     {
58         printf("Real part of number %d: %d\n", i+1,ar[i].Re);
59         printf("Imaginary part of number %d: %d\n", i+1,ar[i].Im);
60     }
61     result=add_complex(ar[0],ar[1]);
62     printf("Real part of number addition: %d\n", result.Re);
63     printf("Imaginary part of number addition: %d\n", result.Im);
64     result=mul_complex(ar[0],ar[1]);
65     printf("Real part of number multiplication: %d\n", result.Re);
66     printf("Imaginary part of number multiplication: %d\n", result.Im);
67 }
68 return 0; //end program
69 }
70
```