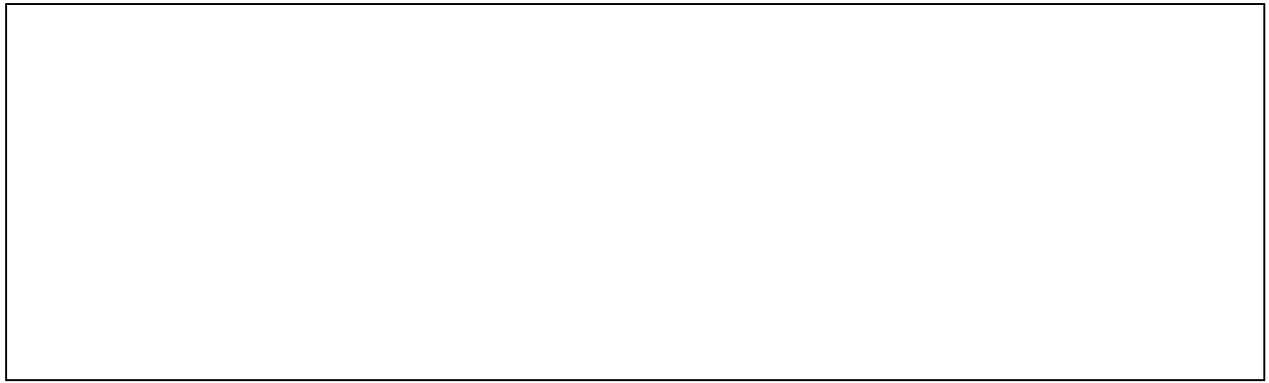


Tutorial 8: Fractions

Problem 1: Fractional Computation

Consider the expression $(17/7)a$. What is the most accurate way to implement this using integer arithmetic in C?



Problem 2: An FIR Filter

Consider the expression for a FIR digital filter in (1):

$$y[n] = \sum_{i=0}^{N-1} a_i x[n-i] \quad (1)$$

$N = 3$, $a_0 = 1.25$, $a_1 = 0.75$, and $a_2 = 2.125$. What is the most accurate way to implement filter this using integer arithmetic in \mathbb{C} ?



Problem 3: Arithmetic Manipulation

Consider the expression $(3/4)b$. By going through the tutorial question 1 you decide to write the C code segment in Figure 6 to represent this expression in the integer arithmetic.

```
int b, d;  
...  
...  
d = (3*b)/4;  
return d;
```

Figure 6: The C-code for $(3/2)b$

Your lab partner understands the content of tutorial question 1 better than you and suggests the code segment in Figure 7.

```
int b, d;  
...  
...  
d = b/2 + b/4;
```

Figure 7: The modified C-code for $3/4b$ - Version 1

You understand the thinking behind your friend's modification and upstage him by suggesting the code in Figure 8.

```
int b, d;  
...  
...  
d = (b + b/2)/2;
```

Figure 8: The modified C-code for $3/4b$ - Version 2

How can you explain the modifications in Figure 7 and Figure 8?

Problem 4: Division by powers of two

Consider the expression $(1/2a + 3/4b + 9/4c)$. Your lab partner going through the tutorial questions 1 and 2 writes the code segment in Figure 9 to represent this expression in integer arithmetic. He uses shift right by 2 ($\gg 2$) for division by 4 ($1/4$).

```
int a, b, c, d;
...
...
d = (2*a + 3*b + 9*c) >> 2;
return d;
}
```

Figure 9: The C-code for $(1/2a + 3/4b + 9/4c)$

Problem 5: Multiplication by 0.7

Write a C code to achieve multiplication by (0.7) using a series of shift sub/add.

