

# Worksheet ADTs

Mike Fraser, Alex Stanhope

Week 10

In this worksheet you will be faced with problems to do with Abstract Data Types.

1. Sketch an ADT for complex numbers with addition, multiplication, and subtraction.

**Answer:**

```
typedef struct Complex Complex ;
Complex *ComplexNew( double real, double imaginary ) ;
void ComplexAdd( Complex *dest, Complex *a, Complex *b);
void ComplexMult(Complex *dest, Complex *a, Complex *b);
void ComplexSub( Complex *dest, Complex *a, Complex *b);
```

2. In the queue explained in the lecture, we use three fields, Front (where we remove elements), Rear (where we add elements) and Count (the number of elements in the queue at present. Assume that the array is of size 25. We initialise Rear, Front and Count to 0.

- (a) What is the value of Front, Count and Rear after we inserted 10 elements?

**Answer:**

0,10,10

- (b) What is the value of Front, Count and Rear when we subsequently take 8 elements out?

**Answer:**

8,10,2

- (c) What is the value of Front, Count and Rear when we insert another 20 elements?

**Answer:**

8,5,22

- (d) What is the relation between Count, Front and Rear? Do you need to maintain all three?

**Answer:**

$\text{Rear} = (\text{Front} + \text{Count}) \bmod 25$ . No, you don't need all three, you can drop one and compute the value from the other two.

3. (a) Sketch an ADT for vectors with vector addition, and vector inner product. The function that creates a new vector should make a vector of  $n$  numbers, where  $n$  is a parameter of the function.

**Answer:**

```
typedef struct Vector Vector ;
Vector *VectorNew( int n ) ;
void VectorSet( Vector *v, int n, double value ) ;
double VectorGet( Vector *v, int n ) ;
void VectorAdd(Vector *dest,Vector *a, Vector *b);
double VectorInnerProd( Vector *a, Vector *b ) ;
```

- (b) What checks must you perform in the functions that add two vectors, and that multiplies two vectors?

**Answer:**

For the matrix vector addition, check that  $n_a = n_b = n_{dest}$ . For the matrix vector inner product, check that  $n_a = n_b$ .

- (c) Sketch an ADT for matrices with matrix times vector multiplication, and matrix matrix multiplication. The function that creates a new matrix should create a  $n$  by  $m$  matrix, where  $n$  and  $m$  are parameters of the function. Use the vector ADT above

**Answer:**

```
#include "vector.h"
typedef struct Matrix Matrix ;
Matrix *MatrixNew( int n, int m ) ;
void MatrixSet(Matrix *v,int n,int o,double value);
double MatrixGet(Matrix *v, int n, int o ) ;
void MatrixVecMult(Vector *dest,Matrix *a,Vector *b);
void MatrixMult(Matrix *dest,Matrix *a,Matrix *b);
```

- (d) What checks must be performed in the two multiplication functions?

**Answer:**

For the matrix multiply, check that  $m_a = n_b$ ,  $n_{dest} = n_a$ ,  $m_{dest} = m_b$ . For the matrix vector multiplication, check that  $m_a = n_b$ ,  $n_{dest} = n_a$ .