Write out your test cases in this document before implementing them in code. Each function requires at least two tests, but possibly more to cover reasonable usage.

Reference the **Function Specifications** section in the main project specification for information on the way the functions work.

Turn in this sheet under the project assignment on Canvas.

# bool IsEmpty() const

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List, do nothing | (none) | IsEmpty() returns true |
| 2. | Create a List, insert 1 item | (none) | IsEmpty() returns false |
| 3. | Create a List, insert 101 items | (none) | IsEmpty() return false |
| 4. | Create a List, insert 100 items | (none) | IsEmpty() return false |
| 5. | Create a List, push 5 items, pop 5 items | (none) | IsEmpty() return true |

# bool IsFull() const

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List, do nothing | (none) | IsFull() returns false |
| 2. | Create a List, insert 100 | (none) | IsFull() returns true |
| 3. | Create a List, insert 101 | (none) | IsFull() returns true |

# int Size() const

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List, do nothing | (none) | Size() returns 0 |
| 2. | Create a List, add 2 items | (none) | Size() returns 2 |
| 3. | Create a List, add 101 items | (none) | Size() returns 100 |
| 4. | Create a List, add 5 items, pop 2 items | (none) | Size() returns 3 |

# int GetCountOf( const T& item ) const

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List, do nothing | “A” | GetCountOf () returns 0 |
| 2. | Create a List, insert 100 items where 75 are “T” | “T” | GetCountOf() returns 75 |
| 3. | Create a List, insert “A”, “B”, “C” | “D” | GetCountOf() returns 0 |
| 4. | Create a List, insert 105 “A”s | “A” | GetCountOf() return 100 |

# bool Contains( const T& item ) const

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List, do nothing | “A” | Contains() returns false |
| 2. | Create a List, insert 100 items where 75 are “T” | “T” | Contains() returns true |
| 3. | Create a List, insert “A”, “B”, “C” | “D” | Contains() returns false |
| 4. | Create a List, insert 105 “A”s | “A” | Contains() returns true |

# List() Constructor

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List | (none) | Size() returns 0. |

# bool PushFront( const T& newItem )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List,  PushFront “A”, “B”, then “C” |  | Get(0) should return “C”  Get(1) should return “B”  Get(2) should return “A” |

# bool PushBack( const T& newItem )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List | “A” | PushBack( ) returns True |
| 2. | Create a List, insert 90 items | “B” | PushBack( ) returns True |
| 3. | Create a List, insert 100 items | “C” (101st item) | PushBack( ) returns False |
| 4. | Create a List, insert 1 item |  | Size() should return 1 |
| 5. | Create a List, insert 90 items |  | Size() should return 90 |
| 6. | Create a List, insert 100 items | “C” (101st item) | Size() should return 100 |
| 7. | Create a List,  PushBack “A”, “B”, then “C” |  | Get(0) should return “A”  Get(1) should return “B”  Get(2) should return “C” |

# bool Insert( int atIndex, const T& item )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List,  PushBack “A”, “B”, “C”  (will be at 0, 1, 2) | 5, “D”  (not contiguous) | Insert() returns false  Size() remains at 3 |
| 2. | Create a List,  Push back “A”, “B”, “C”  (will be at 0, 1, 2) | 1, “D”  (is contiguous) | Size() returns 4  Get(1) returns “D”  Get(2) returns “B”  Get(3) returns “C” |
| 3. | Create a List,  Insert 100 items | 1, “E”  (101st item) | Insert() returns false (it is full)  Size() remains at 100 |

# T\* Get( int atIndex )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  |  |  |

# T\* GetFront()

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  | (none) |  |

# T\* GetBack()

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create List, list is empty | (none) | Nullptr |
| 2. | Create a List, PushBack “a” and “b” | (none) | “b” |

# bool PopFront()

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. | Create a List,  PushBack “A”, “B” | (none) | PopFront() returns true, GetFront() returns “B”, Size() returns 1 |
| 2. | Create a List, PushFront “A”, “B” | (none) | PopFront() returns true, GetFront() returns “A”  Size() returns 1 |
| 3. | Create a List, do nothing | (none) | PopFront() returns false |

# bool PopBack()

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  | (none) |  |

# bool Remove( const T& item )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  |  |  |

# bool Remove( int atIndex )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  |  |  |

# void Clear()

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  | (none) |  |

# bool ShiftRight( int atIndex )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  |  |  |

# bool ShiftLeft( int atIndex )

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **State setup** | **Inputs** | **Expected output** |
| 1. |  |  |  |