5. Array List

List Implementation #2: ______________

```
#pragma once

template <typename T>
class List {
  public:
    /* ... */
  private:
  }
};
```

Array - Implementation Details:

1. What is the running time of []? How?

2. What is the running time of insertFront()? Why?

3. What is the running time of insert()? Why?

4. What is the running time of remove()? Why?

Implementation Details and Analysis:

→ What is our resize strategy?

Array Resize Strategy #1:

...total copies across all resizes: ___________

...total number of insert operations: ___________

...average (amortized) cost of copies per insert: ___________

What is our Big O runtime and amortized runtime?
Array Resize Strategy #2:

...total copies across all resizes: ___________
...total number of insert operations: ___________
...average (amortized) cost of copies per insert: ___________

What is our Big O runtime and amortized runtime?

Running Time:

<table>
<thead>
<tr>
<th></th>
<th>Singly Linked List</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>Look up arbitrary location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insert after a given element</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remove after a given element</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insert at arbitrary location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remove at arbitrary location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search for an input value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Consider tradeoffs between data structures when deciding what to use! Can you think of some ways to improve some of the data structures seen today? What are the consequences?