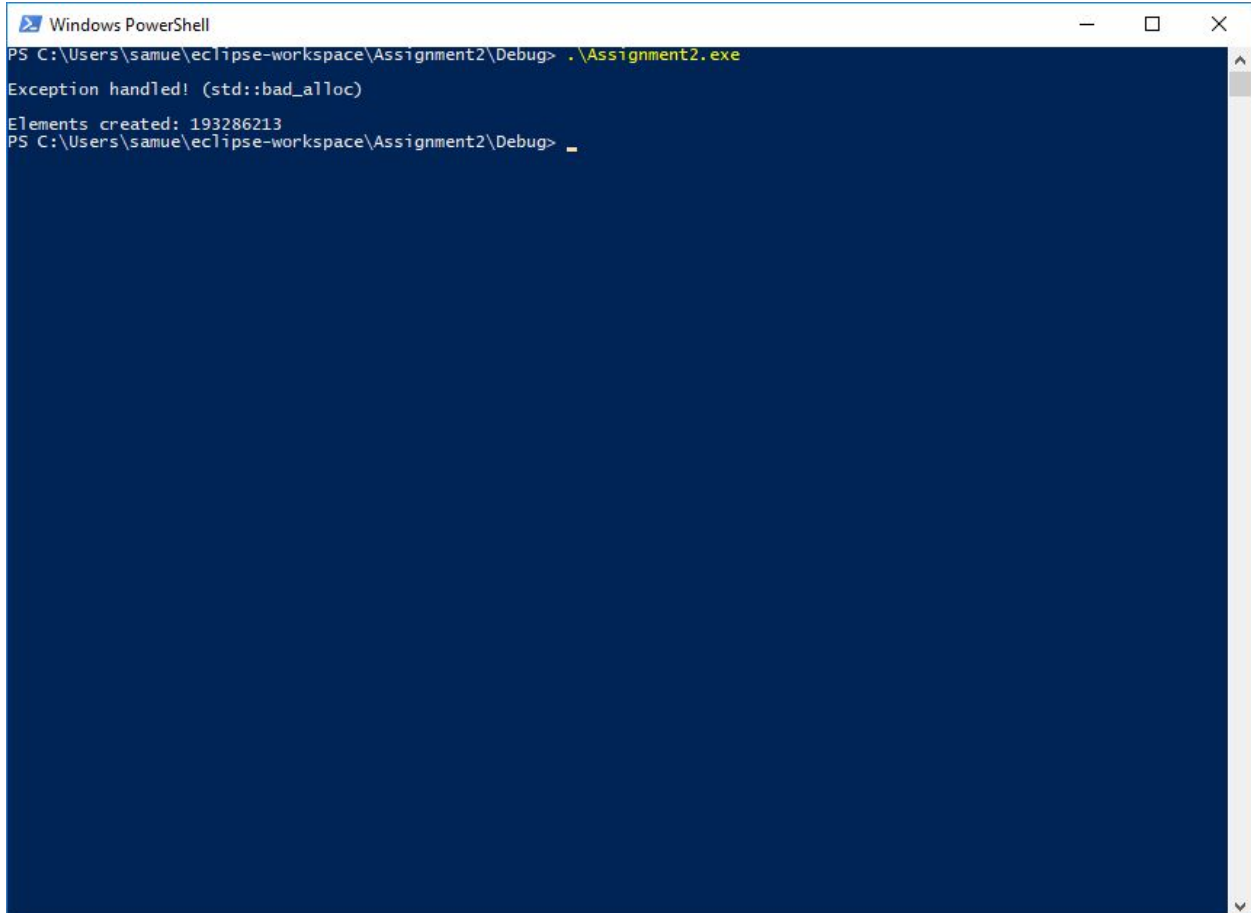


Part A crashed because the system ran out of memory. Using the recursive algorithm, at every new node generated using the “new” operator, more memory is allocated in the heap. Once no more memory is available and a new allocation request is made, a `std::bad_alloc` exception is thrown.

A screenshot of a Windows PowerShell window. The title bar reads "Windows PowerShell". The command prompt shows the user running the command `PS C:\Users\samue\eclipse-workspace\Assignment2\Debug> .\Assignment2.exe`. The output shows "Exception handled! (std::bad_alloc)" followed by "Elements created: 193286213". The prompt then returns to `PS C:\Users\samue\eclipse-workspace\Assignment2\Debug>`.

```
Windows PowerShell
PS C:\Users\samue\eclipse-workspace\Assignment2\Debug> .\Assignment2.exe
Exception handled! (std::bad_alloc)
Elements created: 193286213
PS C:\Users\samue\eclipse-workspace\Assignment2\Debug>
```

```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\samue> cd .\eclipse-workspace\
PS C:\Users\samue\eclipse-workspace> cd .\Assignment2\
PS C:\Users\samue\eclipse-workspace\Assignment2> cd .\Debug\
PS C:\Users\samue\eclipse-workspace\Assignment2\Debug> .\Assignment2.exe

Exception handled! (std::bad_alloc)

PS C:\Users\samue\eclipse-workspace\Assignment2\Debug> .\Assignment2.exe

Exception handled! (std::bad_alloc)

Elements created: 188906220
PS C:\Users\samue\eclipse-workspace\Assignment2\Debug> _
```

But that crash only happened after turning off Windows' automatic management over virtual memory and reducing it. Before doing that, Windows allocated over 23GB to the process during almost one and a half hour of execution before being manually killed.

I was unable to successfully finish the iterative part and all of its tests.

All the tests for Part A and the maximum size of the linked list for recursive functions are in the same file, TestLLis.cpp.