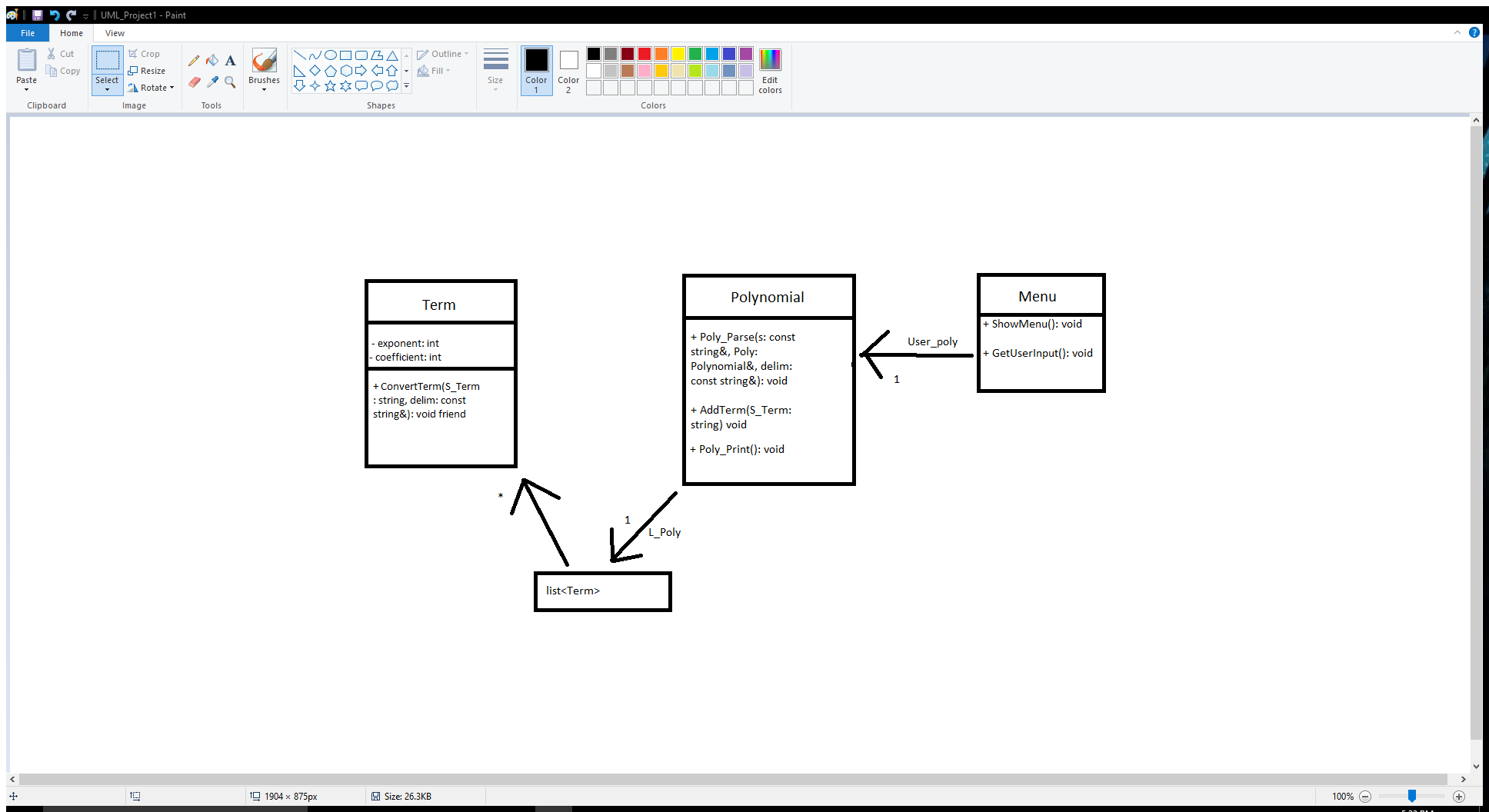
Nathaniel Barnett – Adam Bergman – Urvish Shah

Project 1B

Professor Kuhail

CS 303



Big – O

Poly\_Parse was O(n) because it has to go through every term in polynomial. Can’t be improved because each term requires parsing.

Add\_Term was O(n) because it has to iterate through the list to find out where to add new items in the list. If we could narrow down where it needed to go beforehand it would reduce its cost.

Convert\_Term was O(n) because it walks through each term and modifies it. Since each term needed to be evaluated there isn’t a way to improve its speed.

Print\_Poly was O(n) because we iterate through a list from start to finish. Shouldn’t be changed since each term in a polynomial needs to be printed.

Assumptions:

* User will enter a polynomial without spaces.
* User will not use variables other than x.
* After user has entered two polynomials, if the user wants to enter another two polynomials, the previous set of two polynomials are not to be stored. User wants to enter two polynomials each time they want to add polynomials together.

References

string:: find\_first\_of: http://www.cplusplus.com/reference/string/string/find\_first\_of/

string::find\_first\_not\_of: http://www.cplusplus.com/reference/string/string/find\_first\_not\_of/

Splitting a string: http://ysonggit.github.io/coding/2014/12/16/split-a-string-using-c.html