Connor Perron

One trick mentioned in the thread is the lambda, which is like a function, but it does not have a unique name. You can use this for smaller functions that do not need to be identified many times or needs a specified return type.

You can create a lambda by using the [] operator and follow it with the parameters then adding the body of the lambda in curly braces. An example of a lambda looks like this:

[](int a, double b)

{

return a\*b;

}

Another trick is getting rid of excess include statements but simply using #include<bits/stdc++.h> as it includes vector, time.h, iostream, etc. This makes your code look cleaner and far more professional as it is less "spaghetti like" (multiple noodles of code on one plate, not one long noodle of code).

This is especially useful in BCS230 as well because we cover a wide range of topics and use many different libraries.