

Andrew Nady
900184042

Question 22:

First way:

- I iterate over the numbers from 1 to N
- Where n is the smaller input
- Since I iterate n time, complexity = $O(n)$

```
int gcd(int a, int b)
{
    int n, r;
    if (a > b) n = b;
    else n = a;
    for (int i = 1; i <= n; i++)
    {
        if (a % i == 0 && b % i == 0)
            r = i;
    }

    return r;
}
```

Second Way:

- Here is the recursion form where we used the Euclidean GCD algorithm
- Complexity = $O(\log(\min(a, b))) = O(\log n)$

```
#include <iostream>

#include <iostream>

using namespace std;
int gcd(int a, int b)
{
    if (b == 0)
        return a;
    return gcd(b, a % b);
}
```

Main function:

```
int main()
{
    cout<<gcd(321, 21)<<endl;
    cout<<gcd_S(321, 21);
    return 0;
}
```