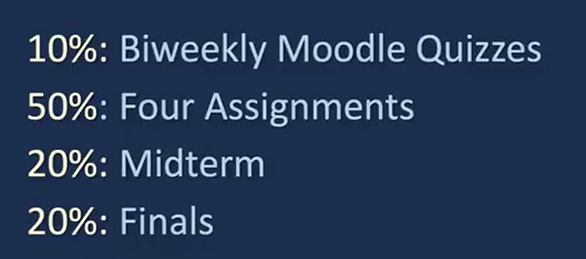
# CS601



<https://www.cse.iitb.ac.in/~rgurjar/CS601_2021/index.php#lectures>

Algorithms: Basic principles like induction/recursion. Basic paradigms like Divide and Conquer, Dynamic Programming, Greedy Algorithms.

Beyond the basics: Bipartite Matching, Network Flow. Reductions.

Complexity: Undecidability, Polynomial time and the Complexity classes NP, co-NP. NP-hardness and NP-completeness.

Randomization: Random variables, Linearity of expectation, Applications like approximate max-cut, quicksort, min-cut.

Pre-requisites: Comfort with abstraction. Basic Run-time analysis, O-notation, Recursion.

Basic probability and combinatorics. Basic data structures -- array, list, stack.

Basic graph algorithms -- cycles, trees, depth-first search, breadth-first search.

Take this [Self Assessment Quiz](https://www.cse.iitb.ac.in/~rgurjar/CS601_2021/assessment.pdf) to check your pre-requisites.

Useful [notes](https://www.cse.iitb.ac.in/~cs601/lec.html) from Prof. Sundar's course. - <https://www.cse.iitb.ac.in/~cs601/lec.html>

Midterm:

So, let's get ready for our Midterm. Here are some of the rules that we will follow in this class for the Midterm.

1. There will be five problems each worth 20 points.
2. Everyone can bring up to two double sided A4 sheets. You can write anything you want on these -- in whatever font size you like. If you want, you can also bring a magnifying glass.
3. If you want, you may write out on the A4 sheets the list of algorithms we have seen in the class. You can use them freely in the midterm without writing any details about these. Of course, if you are modifying these algorithms you need to say clearly how you do it.
4. Remember, we have seen a good number of algorithms so far. For the purposes of the midterm, the important ones are:  
   (Update) Karatsuba, DFS/BFS, Finding connected components in graphs, Topological Sorting of DAGs, Finding SCCs, Dijkstra's Algorithm, Bellman Ford, Fractional Knapsack, Greedy Max Cut, Scheduling, Prim's and Kruskal's
5. Don't be taken aback if some problem worth 20 points appears to have a short solution. The key idea to solve the problem will usually be fairly short. If you can communicate it effectively, you don't have to worry. Note that this does not give you the license to write ambiguous solutions -- this just means you can happily write short solutions which clearly express your arguments.
6. No phones, no laptops, no electonic devies, and no internet is allowed in the exam.
7. (Update) Especially focus on:   
   Induction, Big-Oh, Topological Sort, Divide and Conquer, and Shortest Paths

Final:  
Let me also list what topics you should focus on.

1. Minimum Spanning Trees
2. Dynamic Programming
3. Max Flow
4. NP Completeness
5. Divide and Conquer