RET Computing 2018

Tom Falcone
La Lumiere School
tfalcone@lalumiere.org
La Porte, Indiana (219) 326 7450

RET Impact on My Classes

- AP Computer Science A
  - Programming projects
  - Applications of Computer Science
- AP Statistics
  - Data Acquisition and Analysis
- Intro to Computer Science
  - Python / Applications
    - RUR (Reeborg)

Overview of my work...

- **Researched Graph Coloring problem relating to my work in scheduling students with classes at my school.**

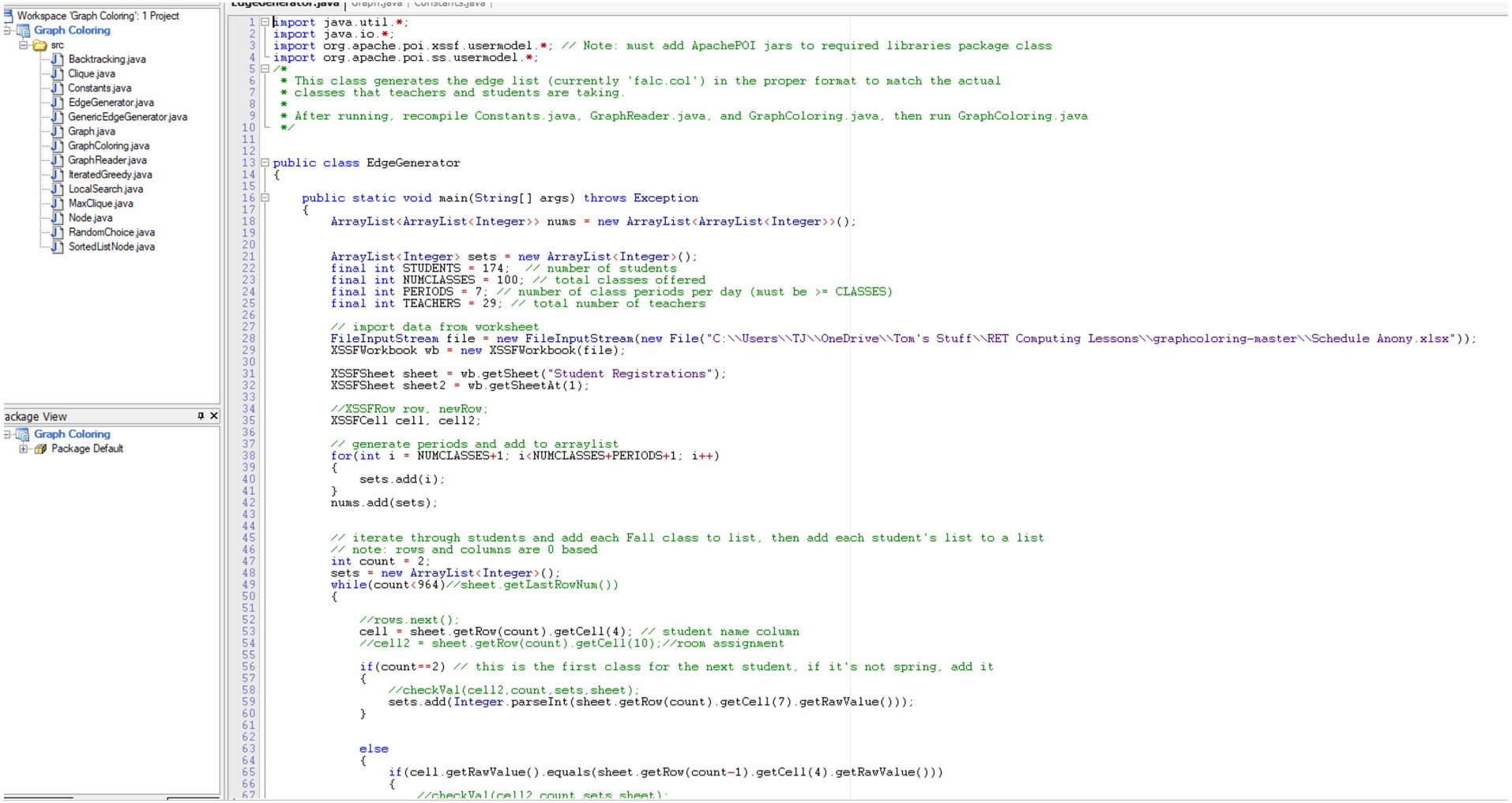
- **Overarching theme in all my studies: Networks!**

My Work

Schedule Allocation:

- The Graph Coloring Problem
  - Graph Theory
  - Brelaz / DSatur Heuristics
  - Apache POI
  - Anonymizing data
  - Limitations

Schedule Allocation



Schedule Allocation

Reading Graph...
Computing Clique...
Maximum Clique Size Found: 9
Vertices in the Clique:
54 48 22 84 2 0 21 7 81

100 milliseconds (excluding I/O).
16 coloring found using DSatur.
Applying Iterated Greedy Improvement...
Found Better Coloring - 15
Found Better Coloring - 14
Applying Local Search...
Final Coloring of graph possible with 14 colors.
Colors of Vertices:
3 9 2 12 3 10 8 13 6 7 3 9 5 12 7
1 12 4 14 10 3 14 8 4 5 13 11 6 3
1 14 8 2 11 1 2 7 13 8 10 4 6 3 10
4 1 4 8 1 10 5 14 11 8 7 11 4 10 3
2 13 7 10 6 1 3 6 3 10 14 1 7 4 2
5 4 3 2 4 11 1 12 5 4 6 10 5 11 1
14 1 13 9 14 2 4 1 12 8 3 8 1 4 7
5 6 2 3

Reading Graph...
Computing Clique...
Maximum Clique Size Found: 7
Vertices in the Clique:
27 30 26 25 24 29 31

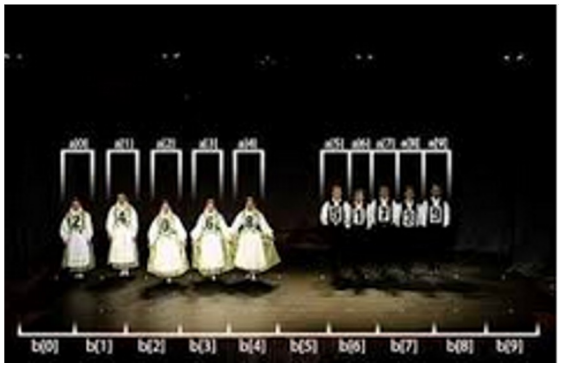
78 milliseconds (excluding I/O).
7 coloring found using DSatur.
Applying Iterated Greedy Improvement...
Applying Local Search...
Final Coloring of graph possible with 7 colors.
Colors of Vertices:
1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1
1 1 1 5 4 3 1 1 6 2 7 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 5 1
1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 7 6 5 4 3
2 1
Process completed.

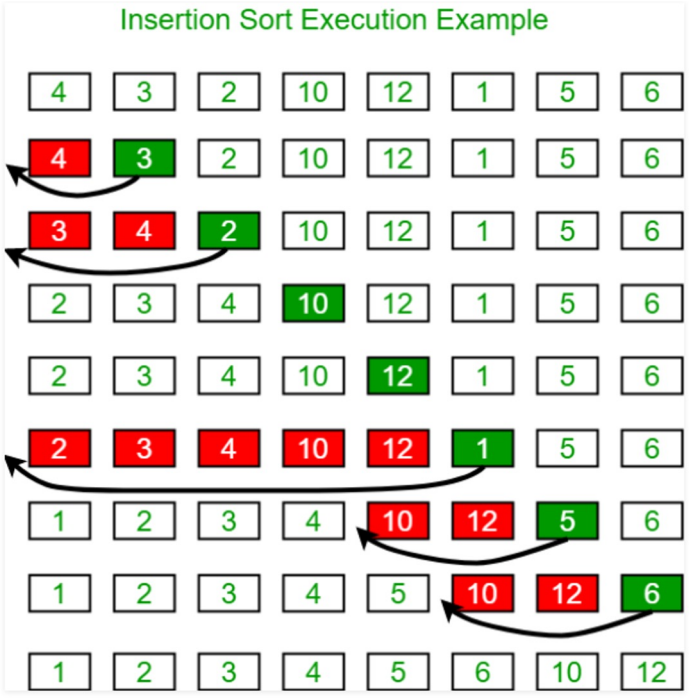
My Work

Sorting Algorithms

- Review the problem from the computer's perspective
- Analyze Insertion, Selection, Quicksort
  - Speed

Sorting Algorithms





Algorithm	Time Complexity			Space Complexity
	Best	Average	Worst	Worst
Quicksort	Ω(n log(n))	Θ(n log(n))	O(n^2)	O(log(n))
Mergesort	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(n)
Timsort	Ω(n)	Θ(n log(n))	O(n log(n))	O(n)
Heapsort	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(1)
Bubble Sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Insertion Sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Selection Sort	Ω(n^2)	Θ(n^2)	O(n^2)	O(1)
Tree Sort	Ω(n log(n))	Θ(n log(n))	O(n^2)	O(n)
Shell Sort	Ω(n log(n))	Θ(n(log(n))^2)	O(n(log(n))^2)	O(1)
Bucket Sort	Ω(n+k)	Θ(n+k)	O(n^2)	O(n)
Radix Sort	Ω(nk)	Θ(nk)	O(nk)	O(n+k)
Counting Sort	Ω(n+k)	Θ(n+k)	O(n+k)	O(k)
Cubesort	Ω(n)	Θ(n log(n))	O(n log(n))	O(n)