Reflect Form
****With parameter of N=100**

PercolationDFS	Time (Mean)
T=100	7.8686
T=200	12.4175
T=400	25.9867
T=800	54.8767
PercolationDFSFast	**Time (Mean)**
T=100	6.0585
T=200	14.0985
T=400	29.0583
T=800	52.3225
PercolationUF (QuickFind)	**Time (Mean)**
T=100	6.7623
T=200	11.3752
T=400	21.0491
T=800	53.3842

PercolationUF (QuickUWPC)	Time (Mean)
T=100	0.2843
T=200	0.5489
T=400	0.8853
T=800	1.2238

****With parameter of T=50**

PercolationDFS	Time (Mean)
N=25	0.0698
N=50	0.2321
N=100	3.2763
N=200	37.7658

PercolationDFSFast	Time (Mean)
N=25	0.1837
N=50	0.2768
N=100	5.9847
N=200	45.1022
PercolationUF (QuickFind)	**Time (Mean)**
N=25	0.0294
N=50	0.1827
N=100	2.9842
N=200	45.2983
PercolationUF (QuickUWPC)	**Time (Mean)**
N=25	0.0294
N=50	0.0674
N=100	0.1823
N=200	0.2765

PercolationDFS

1. How does doubling the grid-size, N, affect the running time?

Increases by approximately 2^4.

2. How does doubling the number of experiments, T, performed affect the running time?

It increases linearly, and doubles. (approximately)

3. Try to provide a formula for the running time in terms of N and T, use big-Oh.

*O(T) * O(N^4) due to the complexity.*

4. Estimate the largest grid-size you can run in a day for 100 trials (assume time is the only limit here, not memory).

When both N and T are 100, the runtime is 7.8686. There are 86,400 seconds in a day. Solving for N, the largest grid-size would be approximately 1200.

5. Give estimate for how much memory is used in terms of N, the grid-size. Provide your estimate in bytes and use four bytes for an int, one byte for a boolean, and eight bytes for a double. For example, an array of N integers uses 4N bytes in this model, there's no overhead for the array other than storing the integer values.

(*N^2)*

PercolationDFSFast

1. How does doubling the grid-size, N, affect the running time?

By a factor of four, or quadruples it.

2. How does doubling the number of experiments, T, performed affect the running time?

It doubles the runtime.

3. Try to provide a formula for the running time in terms of N and T, use big-Oh.

*O(T) * O(N^2)*

4. Estimate the largest grid-size you can run in a day for 100 trials (assume time is the only limit here, not memory).

The runtime when both N and T are 100 is 6.0585. The runtime formula can be found in t(N/100)^2. When we solve for N, this gives us approximately 60,000.*

5. Give estimate for how much memory is used in terms of N, the grid-size. Provide your estimate in bytes and use four bytes for an int, one byte for a boolean, and eight bytes for a double. For example, an array of N integers uses 4N bytes in this model, there's no overhead for the array other than storing the integer values.

*4 * N^2*

PercolationUF with QuickFind

1. How does doubling the grid-size, N, affect the running time?

Quadrupling (approximately).

2. How does doubling the number of experiments, T, performed affect the running time?

Runtime is also doubled.

3. Try to provide a formula for the running time in terms of N and T, use big-Oh.

O(T) + O(N^4) to account for both the grid size and number of experiments.

4. Estimate the largest grid-size you can run in a day for 100 trials (assume time is the only limit here, not memory).

A day has 86,400 seconds. The runtime formula must be t(N/100) ^ 4. If you solve for N, you get approximately 1200.*

5. Give estimate for how much memory is used in terms of N, the grid-size. Provide your estimate in bytes and use four bytes for an int, one byte for a boolean, and eight bytes for a double. For example, an array of N integers uses 4N bytes in this model, there's no overhead for the array other than storing the integer values.

N^2 (boolean array) + 4(N^2+2) (integer array for indices)

PercolationUF with QuickUWPC

1. How does doubling the grid-size, N, affect the running time?

When I switched the grid-size from 5 to 10, the run time went from 0.0041 to 0.0107, an increase of approximately 0.0060. When I doubled the size again, run time increased to 0.147, an increase of approximately 0.0080. This is a linear increase, and is generally doubling.

2. How does doubling the number of experiments, T, performed affect the running time?

With 5 trials, the run time is 0.0036. When I doubled the trials to 20, the runtime increased to 0.0049, increasing very minimally. When I doubled the trials again to 40, the runtime increased to 0.0073. After doubling again, it became 0.0108. This shows a steady linear increase of doubling.

3. Try to provide a formula for the running time in terms of N and T, use big-Oh.

O((T)(N^2))*

4. Estimate the largest grid-size you can run in a day for 100 trials (assume time is the only limit here, not memory).

I believe that this would be approximately 60,000. There are 86,400 seconds in a day, and if N and T are both 100, QuickUWPC runtime is 0.2843. The formula would be t(N/100)^2.*

5. Give estimate for how much memory is used in terms of N, the grid-size. Provide your estimate in bytes and use four bytes for an int, one byte for a boolean, and eight bytes for a

double. For example, an array of N integers uses 4N bytes in this model, there's no overhead for the array other than storing the integer values.

N^2 (boolean array) + 4(N^2+2) (integer array for indices)