Compsci 201: DNA Analysis Callie Mao

Part I: Benchmarking cutAndSplice with SimpleStrand

I tested SimpleStrand.cutAndSplice utilizing several files of varying lengths. In order to determine the runtime of cutAndSplice, different recombinant sizes would have to be tested and the time to generate each recombinant measured. Different file lengths are used to ensure that this there is no effect of DNA length on the results.

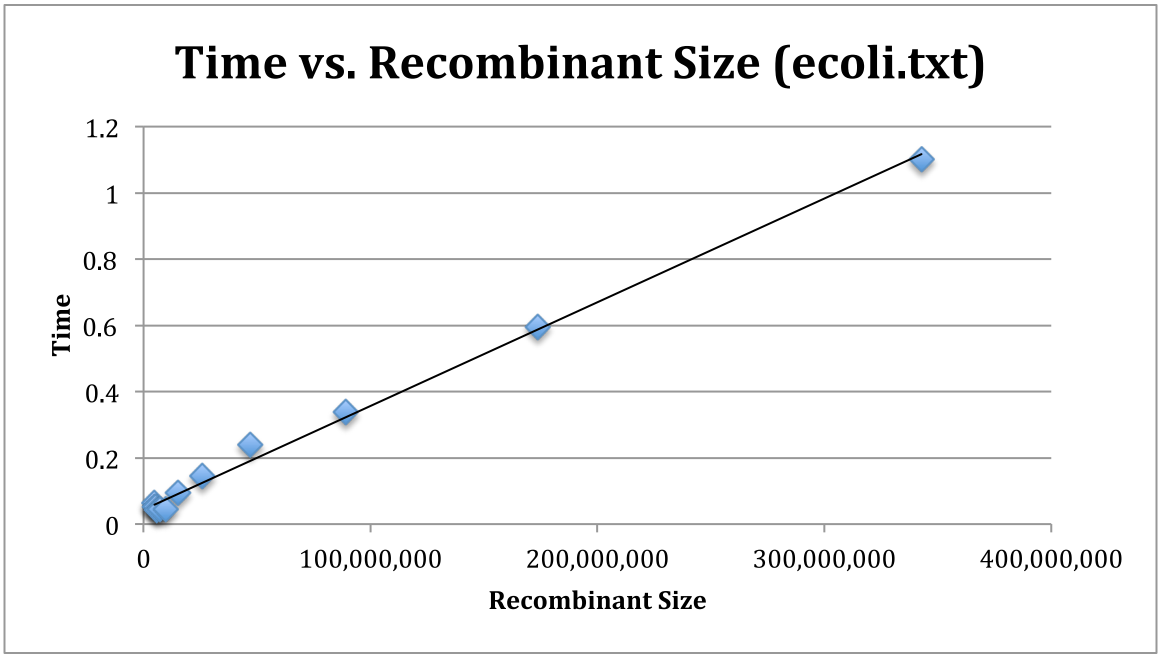
The given ecoli.txt and ecoli_small.txt, with 4,639,221 and 320,160 in DNA length respectively, along with ecoli_double.txt, which I created to double the length of the original ecoli.txt with DNA length of 9,278,442, were utilized to test for the relationship between recombinant size and time. The results (shown below), display a linear relationship between the size of the recombinant strand and the time taken to execute the cutAndSplice method. In each trial, the length of the DNA (deterimed by the file) is held constant. Regardless of DNA length, the same linear relationship holds true, and thus suggests O(N) behavior.

Below are the results from ecoli.txt, ecoli_small.txt, and ecoli_double.txt from cutting at enzyme gaattc:

ecoli.txt

dna length = 4,639,221

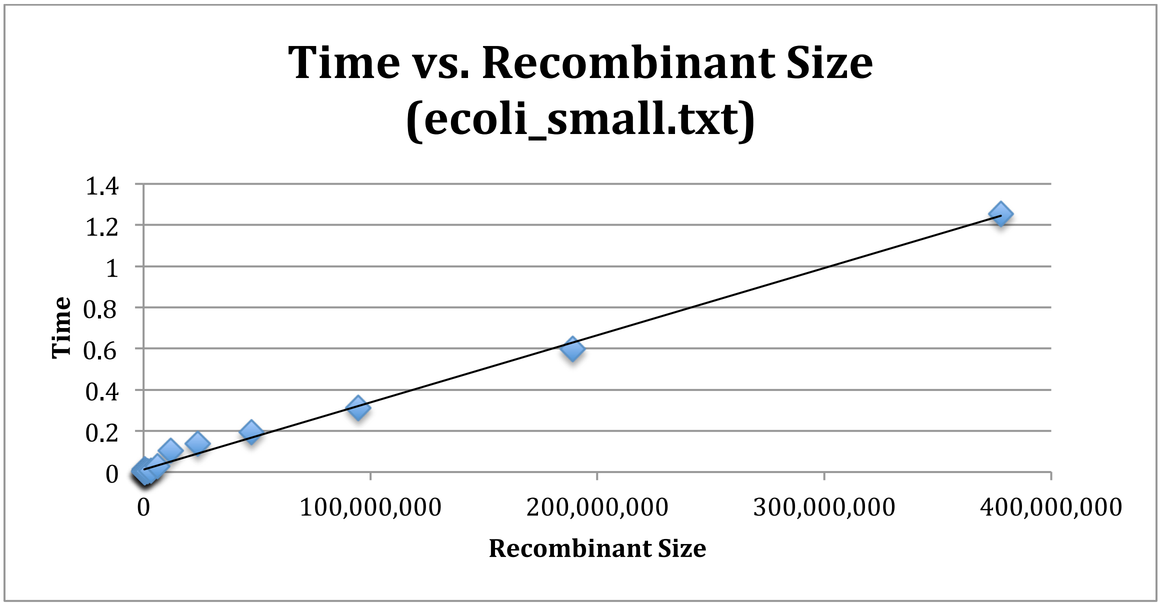
Class	splicee	recomb	time	
SimpleStrand:	256	4,800,471	0.064	# append calls = 1290
SimpleStrand:	512	4,965,591	0.053	# append calls = 1290
SimpleStrand:	1,024	5,295,831	0.052	# append calls = 1290
SimpleStrand:	2,048	5,956,311	0.042	# append calls = 1290
SimpleStrand:	4,096	7,277,271	0.044	# append calls = 1290
SimpleStrand:	8,192	9,919,191	0.044	# append calls = 1290
SimpleStrand:	16,384	15,203,031	0.094	# append calls = 1290
SimpleStrand:	32,768	25,770,711	0.146	# append calls = 1290
SimpleStrand:	65,536	46,906,071	0.239	# append calls = 1290
SimpleStrand:	131,072	89,176,791	0.338	# append calls = 1290
SimpleStrand:	262,144	173,718,231	0.596	# append calls = 1290
SimpleStrand:	524,288	342,801,111	1.102	# append calls = 1290



ecoli_small

dna length = 320,160

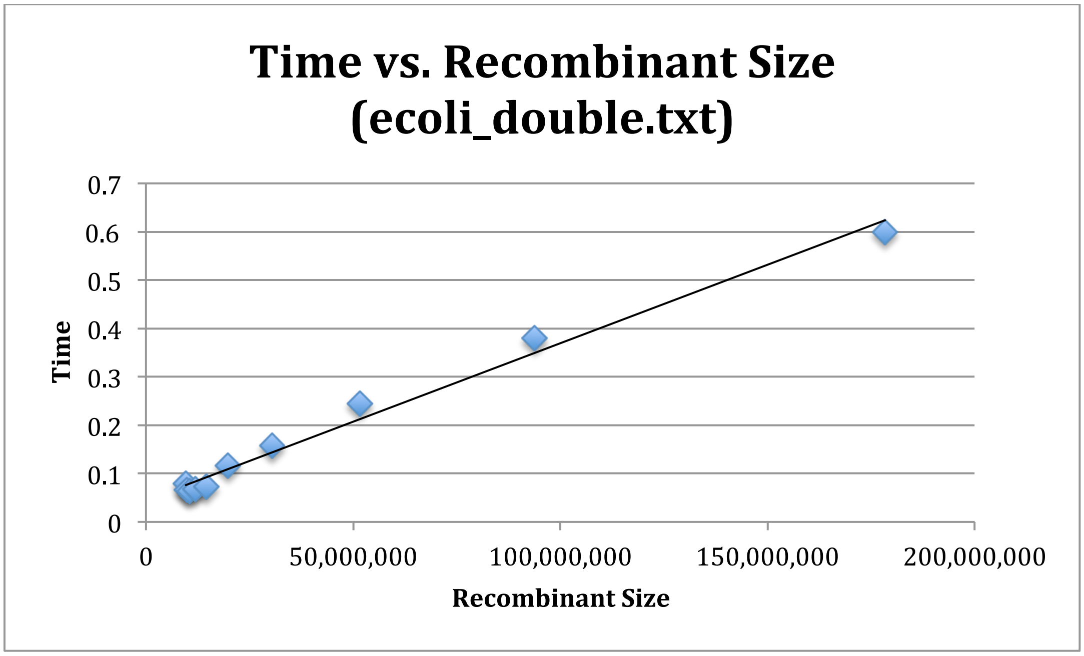
Class	splicee	recomb	time	
SimpleStrand:	256	331,410	0.003	# append calls = 90
SimpleStrand:	512	342,930	0.002	# append calls = 90
SimpleStrand:	1,024	365,970	0.014	# append calls = 90
SimpleStrand:	2,048	412,050	0.002	# append calls = 90
SimpleStrand:	4,096	504,210	0.003	# append calls = 90
SimpleStrand:	8,192	688,530	0.003	# append calls = 90
SimpleStrand:	16,384	1,057,170	0.005	# append calls = 90
SimpleStrand:	32,768	1,794,450	0.004	# append calls = 90
SimpleStrand:	65,536	3,269,010	0.008	# append calls = 90
SimpleStrand:	131,072	6,218,130	0.029	# append calls = 90
SimpleStrand:	262,144	12,116,370	0.103	# append calls = 90
SimpleStrand:	524,288	23,912,850	0.138	# append calls = 90
SimpleStrand:	1,048,576	47,505,810	0.193	# append calls = 90
SimpleStrand:	2,097,152	94,691,730	0.311	# append calls = 90
SimpleStrand:	4,194,304	189,063,570	0.599	# append calls = 90
SimpleStrand:	8,388,608	377,807,250	1.255	# append calls = 90



ecoli_double.txt

dna length = 9,278,442

Class	splicee	recomb	time	
SimpleStrand:	256	9,600,942	0.079	# append calls = 2580
SimpleStrand:	512	9,931,182	0.066	# append calls = 2580
SimpleStrand:	1,024	10,591,662	0.063	# append calls = 2580
SimpleStrand:	2,048	11,912,622	0.067	# append calls = 2580
SimpleStrand:	4,096	14,554,542	0.073	# append calls = 2580
SimpleStrand:	8,192	19,838,382	0.117	# append calls = 2580
SimpleStrand:	16,384	30,406,062	0.158	# append calls = 2580
SimpleStrand:	32,768	51,541,422	0.245	# append calls = 2580
SimpleStrand:	65,536	93,812,142	0.38	# append calls = 2580
SimpleStrand:	131,072	178,353,582	0.599	# append calls = 2580



Part II: Benchmarking Memory

Largest splice length generated and the time taken were obtained for the following memory sizes: 128M, 256M, 512M, 1024M. A linear, O(N) relationship was found in respect to both largest splice length and time taken.. When the heap size doubles, the splice length and time taken approximately double. Because increasing memory allows the program to run through more data before memory exhaustion, an increase in runtime and splice length is obtained. Largest splice and time range from 16,384 at 0.082 seconds with 128M to 65,536 at 0.203 seconds with 256M all the way to 262, 144 at 0.58 seconds with 1024M. Doubling the heap size every time allows my machine to generate the next power-of-two strand. The data below demonstrates these linear increases using ecoli.txt (keeping the DNA length constant in each trial at 4,639,221) and cutting at enzyme gaattc:

128 M

```
dna length = 4,639,221

Class          splicee          recomb        time
SimpleStrand:        256       4,800,471      0.056   # append calls = 1290
SimpleStrand:        512       4,965,591      0.056   # append calls = 1290
SimpleStrand:      1,024       5,295,831      0.047   # append calls = 1290
SimpleStrand:      2,048       5,956,311      0.041   # append calls = 1290
SimpleStrand:      4,096       7,277,271      0.085   # append calls = 1290
SimpleStrand:      8,192       9,919,191      0.043   # append calls = 1290
SimpleStrand:     16,384      15,203,031      0.082   # append calls = 1290
```

256 M

```
dna length = 4,639,221

Class          splicee          recomb        time
SimpleStrand:        256       4,800,471      0.052   # append calls = 1290
SimpleStrand:        512       4,965,591      0.051   # append calls = 1290
SimpleStrand:      1,024       5,295,831      0.043   # append calls = 1290
SimpleStrand:      2,048       5,956,311      0.045   # append calls = 1290
SimpleStrand:      4,096       7,277,271      0.041   # append calls = 1290
SimpleStrand:      8,192       9,919,191      0.087   # append calls = 1290
SimpleStrand:     16,384      15,203,031      0.083   # append calls = 1290
SimpleStrand:     32,768      25,770,711      0.135   # append calls = 1290
SimpleStrand:     65,536      46,906,071      0.203   # append calls = 1290
```

512 M

```
dna length = 4,639,221

Class          splicee          recomb        time
SimpleStrand:        256       4,800,471      0.056   # append calls = 1290
```

SimpleStrand:	512	4,965,591	0.048	# append calls = 1290
SimpleStrand:	1,024	5,295,831	0.081	# append calls = 1290
SimpleStrand:	2,048	5,956,311	0.045	# append calls = 1290
SimpleStrand:	4,096	7,277,271	0.048	# append calls = 1290
SimpleStrand:	8,192	9,919,191	0.089	# append calls = 1290
SimpleStrand:	16,384	15,203,031	0.128	# append calls = 1290
SimpleStrand:	32,768	25,770,711	0.15	# append calls = 1290
SimpleStrand:	65,536	46,906,071	0.205	# append calls = 1290
SimpleStrand:	**131,072**	89,176,791	**0.34**	# append calls = 1290

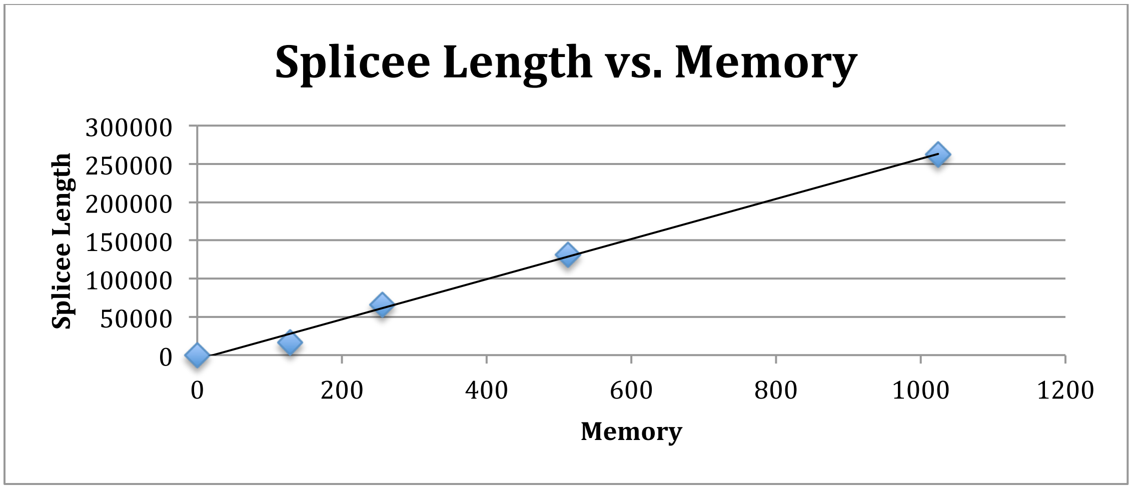
1024M

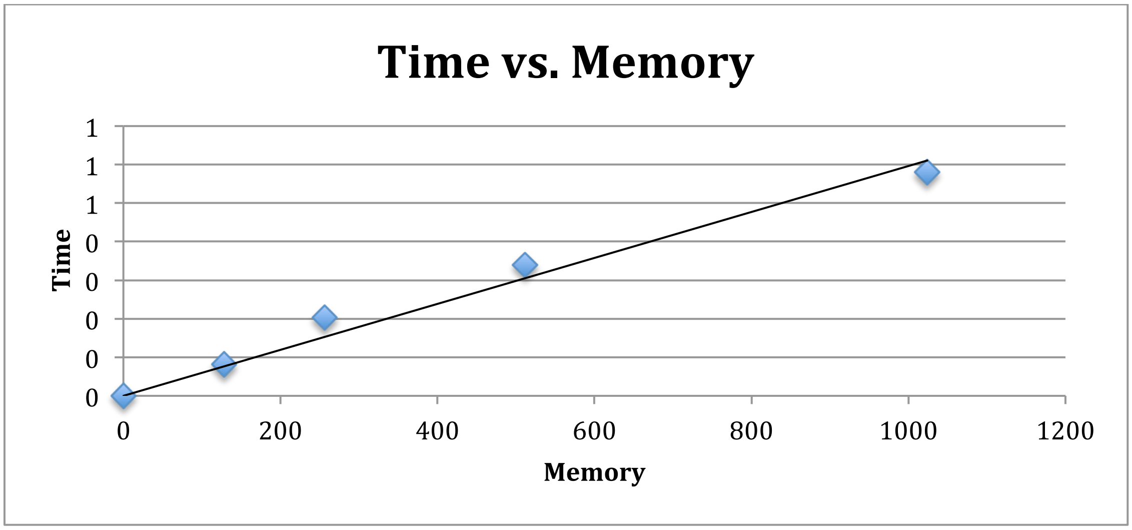
dna length = 4,639,221

Class	splicee	recomb	time	
SimpleStrand:	256	4,800,471	0.056	# append calls = 1290
SimpleStrand:	512	4,965,591	0.055	# append calls = 1290
SimpleStrand:	1,024	5,295,831	0.05	# append calls = 1290
SimpleStrand:	2,048	5,956,311	0.041	# append calls = 1290
SimpleStrand:	4,096	7,277,271	0.041	# append calls = 1290
SimpleStrand:	8,192	9,919,191	0.048	# append calls = 1290
SimpleStrand:	16,384	15,203,031	0.101	# append calls = 1290
SimpleStrand:	32,768	25,770,711	0.151	# append calls = 1290
SimpleStrand:	65,536	46,906,071	0.215	# append calls = 1290
SimpleStrand:	131,072	89,176,791	0.343	# append calls = 1290
SimpleStrand:	**262,144**	173,718,231	**0.58**	# append calls = 1290

Compiling the data above (bolded) into a comprehensive table, the following results were obtained below.

Memory	**Splicee Length**	**Time**
128	16,384	0.082
256	65,536	0.203
512	131,072	0.34
1024	262,144	0.58





Part III: Benchmarking LinkStrand

Several different ecoli texts were used to measure the relationship between time and the number of breaks. The original ecoli.txt and ecoli_small.txt were used, along with ecoli_double.txt (double the original ecoli.txt), ecoli_quadruple.txt (quadruple the original ecoli.txt), and ecoli50000.txt (the first 50000 lines of ecoli.txt) that I created.

The number of append calls was measured for each ecoli document and was divided by 2 to get the number of breaks. This is done because there are 2 append calls for each call to the cutAndSplice method for just 1 break. The time taken was averaged among the times within the document. After doing so, the resulting relationship was graphed (table and graph at the bottom of the below data) for time taken and the number of breaks. The relationship between them was a linear, O(N) graph, or in this case, O(B) where B is the number of breaks. Below is the data supporting this conclusion, with the bolded values used in the final table and graph:

ecoli_quadruple.txt

dna length = 18,556,884

Class	splicee	recomb	time	
LinkStrand:	256	19,201,884	0.229	**# append calls = 5160**
LinkStrand:	512	19,862,364	0.157	**# append calls = 5160**
LinkStrand:	1,024	21,183,324	0.14	**# append calls = 5160**
LinkStrand:	2,048	23,825,244	0.151	**# append calls = 5160**
LinkStrand:	4,096	29,109,084	0.116	**# append calls = 5160**
LinkStrand:	8,192	39,676,764	0.153	**# append calls = 5160**
LinkStrand:	16,384	60,812,124	0.108	**# append calls = 5160**
LinkStrand:	32,768	103,082,844	0.148	**# append calls = 5160**
LinkStrand:	65,536	187,624,284	0.11	**# append calls = 5160**
LinkStrand:	131,072	356,707,164	0.158	**# append calls = 5160**
LinkStrand:	262,144	694,872,924	0.111	**# append calls = 5160**
LinkStrand:	524,288	1,371,204,444	0.151	**# append calls = 5160**
LinkStrand:	1,048,576	2,723,867,484	0.114	**# append calls = 5160**
LinkStrand:	2,097,152	5,429,193,564	0.144	**# append calls = 5160**
LinkStrand:	4,194,304	10,839,845,724	0.16	**# append calls = 5160**
LinkStrand:	8,388,608	21,661,150,044	0.151	**# append calls = 5160**
LinkStrand:	16,777,216	43,303,758,684	0.15	**# append calls = 5160**
LinkStrand:	33,554,432	86,588,975,964	0.16	**# append calls = 5160**
LinkStrand:	67,108,864	173,159,410,524	0.182	**# append calls = 5160**
LinkStrand:	134,217,728	346,300,279,644	0.131	**# append calls = 5160**
		Time Average	0.153894737	

ecoli_double.txt

dna length = 9,278,442

Class	splicee	recomb	time	
LinkStrand:	256	9,600,942	0.094	**# append calls = 2580**
LinkStrand:	512	9,931,182	0.062	**# append calls = 2580**
LinkStrand:	1,024	10,591,662	0.095	**# append calls = 2580**
LinkStrand:	2,048	11,912,622	0.052	**# append calls = 2580**
LinkStrand:	4,096	14,554,542	0.089	**# append calls = 2580**
LinkStrand:	8,192	19,838,382	0.058	**# append calls = 2580**
LinkStrand:	16,384	30,406,062	0.107	**# append calls = 2580**
LinkStrand:	32,768	51,541,422	0.057	**# append calls = 2580**
LinkStrand:	65,536	93,812,142	0.106	**# append calls = 2580**
LinkStrand:	131,072	178,353,582	0.053	**# append calls = 2580**
LinkStrand:	262,144	347,436,462	0.094	**# append calls = 2580**
LinkStrand:	524,288	685,602,222	0.055	**# append calls = 2580**
LinkStrand:	1,048,576	1,361,933,742	0.092	**# append calls = 2580**

LinkStrand:	2,097,152	2,714,596,782	0.052	**# append calls = 2580**
LinkStrand:	4,194,304	5,419,922,862	0.088	**# append calls = 2580**
LinkStrand:	8,388,608	10,830,575,022	0.05	**# append calls = 2580**
LinkStrand:	16,777,216	21,651,879,342	0.097	**# append calls = 2580**
LinkStrand:	33,554,432	43,294,487,982	0.102	**# append calls = 2580**
LinkStrand:	67,108,864	86,579,705,262	0.064	**# append calls = 2580**
LinkStrand:	134,217,728	173,150,139,822	0.055	**# append calls = 2580**
		Time Average	**0.080105263**	

ecoli

dna length = 4,639,221

Class	splicee	recomb	time	
LinkStrand:	256	4,800,471	0.047	**# append calls = 1290**
LinkStrand:	512	4,965,591	0.034	**# append calls = 1290**
LinkStrand:	1,024	5,295,831	0.034	**# append calls = 1290**
LinkStrand:	2,048	5,956,311	0.034	**# append calls = 1290**
LinkStrand:	4,096	7,277,271	0.031	**# append calls = 1290**
LinkStrand:	8,192	9,919,191	0.036	**# append calls = 1290**
LinkStrand:	16,384	15,203,031	0.033	**# append calls = 1290**
LinkStrand:	32,768	25,770,711	0.03	**# append calls = 1290**
LinkStrand:	65,536	46,906,071	0.031	**# append calls = 1290**
LinkStrand:	131,072	89,176,791	0.03	**# append calls = 1290**
LinkStrand:	262,144	173,718,231	0.032	**# append calls = 1290**
LinkStrand:	524,288	342,801,111	0.03	**# append calls = 1290**
LinkStrand:	1,048,576	680,966,871	0.033	**# append calls = 1290**
LinkStrand:	2,097,152	1,357,298,391	0.03	**# append calls = 1290**
LinkStrand:	4,194,304	2,709,961,431	0.035	**# append calls = 1290**
LinkStrand:	8,388,608	5,415,287,511	0.034	**# append calls = 1290**
LinkStrand:	16,777,216	10,825,939,671	0.036	**# append calls = 1290**
LinkStrand:	33,554,432	21,647,243,991	0.035	**# append calls = 1290**
LinkStrand:	67,108,864	43,289,852,631	0.023	**# append calls = 1290**
LinkStrand:	134,217,728	86,575,069,911	0.024	**# append calls = 1290**
		Time Average	**0.034315789**	

ecoli50000

dna length = 3,000,000

Class	splicee	recomb	time	
LinkStrand:	256	3,102,000	0.025	**# append calls = 816**
LinkStrand:	512	3,206,448	0.018	**# append calls = 816**
LinkStrand:	1,024	3,415,344	0.021	**# append calls = 816**
LinkStrand:	2,048	3,833,136	0.017	**# append calls = 816**
LinkStrand:	4,096	4,668,720	0.02	**# append calls = 816**

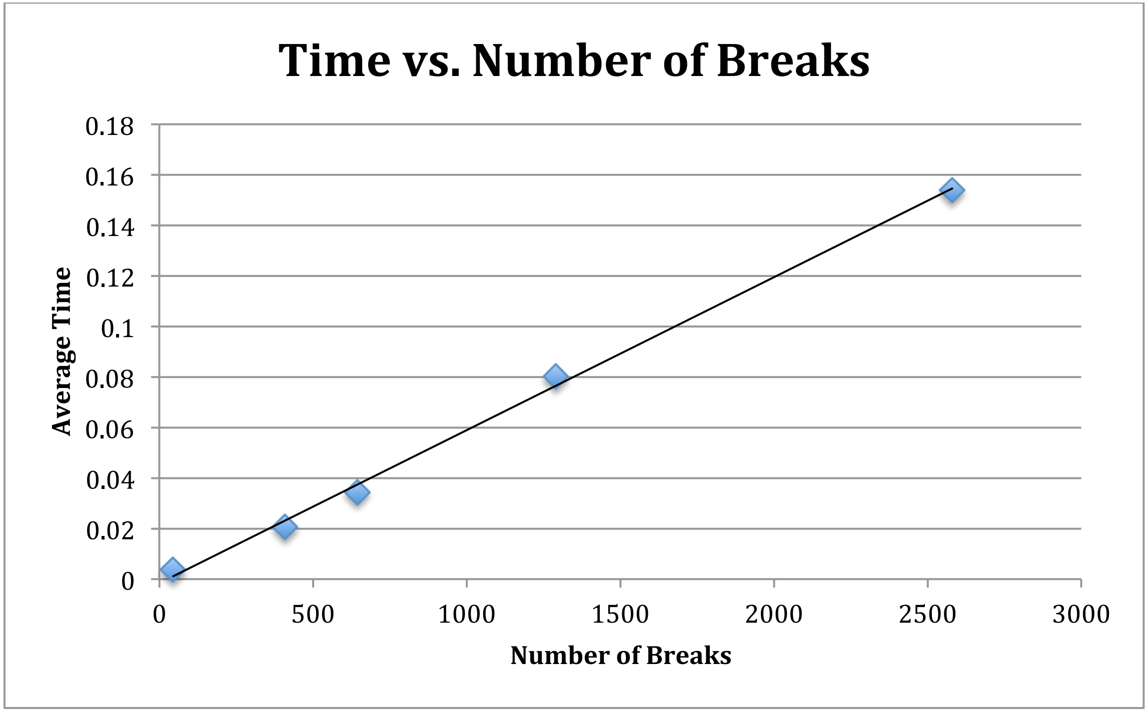
LinkStrand:	8,192	6,339,888	0.023	**# append calls = 816**
LinkStrand:	16,384	9,682,224	0.019	**# append calls = 816**
LinkStrand:	32,768	16,366,896	0.018	**# append calls = 816**
LinkStrand:	65,536	29,736,240	0.019	**# append calls = 816**
LinkStrand:	131,072	56,474,928	0.019	**# append calls = 816**
LinkStrand:	262,144	109,952,304	0.018	**# append calls = 816**
LinkStrand:	524,288	216,907,056	0.019	**# append calls = 816**
LinkStrand:	1,048,576	430,816,560	0.017	**# append calls = 816**
LinkStrand:	2,097,152	858,635,568	0.026	**# append calls = 816**
LinkStrand:	4,194,304	1,714,273,584	0.022	**# append calls = 816**
LinkStrand:	8,388,608	3,425,549,616	0.024	**# append calls = 816**
LinkStrand:	16,777,216	6,848,101,680	0.017	**# append calls = 816**
LinkStrand:	33,554,432	13,693,205,808	0.017	**# append calls = 816**
LinkStrand:	67,108,864	27,383,414,064	0.016	**# append calls = 816**
LinkStrand:	134,217,728	54,763,830,576	0.018	**# append calls = 816**
		Time Average	**0.020684211**	

ecoli_small

dna length = 320,160

Class	splicee	recomb	time	
LinkStrand:	256	331,410	0.002	**# append calls = 90**
LinkStrand:	512	342,930	0.003	**# append calls = 90**
LinkStrand:	1,024	365,970	0.001	**# append calls = 90**
LinkStrand:	2,048	412,050	0.002	**# append calls = 90**
LinkStrand:	4,096	504,210	0.002	**# append calls = 90**
LinkStrand:	8,192	688,530	0.022	**# append calls = 90**
LinkStrand:	16,384	1,057,170	0.002	**# append calls = 90**
LinkStrand:	32,768	1,794,450	0.002	**# append calls = 90**
LinkStrand:	65,536	3,269,010	0.002	**# append calls = 90**
LinkStrand:	131,072	6,218,130	0.002	**# append calls = 90**
LinkStrand:	262,144	12,116,370	0.002	**# append calls = 90**
LinkStrand:	524,288	23,912,850	0.001	**# append calls = 90**
LinkStrand:	1,048,576	47,505,810	0.002	**# append calls = 90**
LinkStrand:	2,097,152	94,691,730	0.003	**# append calls = 90**
LinkStrand:	4,194,304	189,063,570	0.002	**# append calls = 90**
LinkStrand:	8,388,608	377,807,250	0.012	**# append calls = 90**
LinkStrand:	16,777,216	755,294,610	0.003	**# append calls = 90**
LinkStrand:	33,554,432	1,510,269,330	0.002	**# append calls = 90**
LinkStrand:	67,108,864	3,020,218,770	0.002	**# append calls = 90**
LinkStrand:	134,217,728	6,040,117,650	0.002	**# append calls = 90**
		Time Average	**0.003736842**	

File	# Append Calls	# Breaks	Average Time
ecoli_quadruple.txt	5160	2580	0.153894737
ecoli_double.txt	2580	1290	0.080105263
ecoli.txt	1290	645	0.034315789
ecoli50000.txt	816	408	0.020684211
ecoli_small.txt	90	45	0.003736842



Extra Credit

I utilized a HashMap to save each unique string that was already reversed. The key of the HashMap pointed to each unique string and its value to its reversed string. HashMap is O(1) for adding in a key-value pair and retrieving the reversed string and is thus fairly efficient. Thus, for each time a string that was already contained in the HashMap was called, the HashMap could simply retrieve the already reversed string. If the string was new and not used before (and thus not in the HashMap), the string could be reversed. Doing this ensures that each unique string is only reversed once and the reverse string can be retrieved should there be any duplicates.