

Data:
 dna length = 4,639,221
 cutting at enzyme gaattc

Class	splicee	recomb	time	appends

StringBuilderStrand:	256	4,800,471	0.037	1290
StringBuilderStrand:	512	4,965,591	0.024	1290
StringBuilderStrand:	1,024	5,295,831	0.010	1290
StringBuilderStrand:	2,048	5,956,311	0.009	1290
StringBuilderStrand:	4,096	7,277,271	0.008	1290
StringBuilderStrand:	8,192	9,919,191	0.012	1290
StringBuilderStrand:	16,384	15,203,031	0.020	1290
StringBuilderStrand:	32,768	25,770,711	0.030	1290
StringBuilderStrand:	65,536	46,906,071	0.040	1290
StringBuilderStrand:	131,072	89,176,791	0.123	1290
StringBuilderStrand:	262,144	173,718,231	0.118	1290
StringBuilderStrand:	524,288	342,801,111	0.322	1290
StringBuilderStrand:	1,048,576	680,966,871	0.556	1290

dna length = 4,639,221
 cutting at enzyme gaattc

Class	splicee	recomb	time	appends

StringStrand:	256	4,800,471	0.452	1290
StringStrand:	512	4,965,591	0.488	1290
StringStrand:	1,024	5,295,831	0.540	1290
StringStrand:	2,048	5,956,311	0.573	1290
StringStrand:	4,096	7,277,271	0.733	1290
StringStrand:	8,192	9,919,191	1.087	1290
StringStrand:	16,384	15,203,031	1.880	1290
StringStrand:	32,768	25,770,711	3.385	1290
StringStrand:	65,536	46,906,071	6.496	1290
StringStrand:	131,072	89,176,791	12.582	1290
StringStrand:	262,144	173,718,231	25.851	1290
StringStrand:	524,288	342,801,111	50.184	1290
StringStrand:	1,048,576	680,966,871	109.968	1290
StringStrand:	2,097,152	1,357,298,391	4384.087	1290

Questions:

1. Are the benchmark timings for StringStrand consistent with the explanation below that the time to execute cutAndSplice is $O(b^2S)$?

Looking at the data for StringStrand, append remains constant at 1290. Therefore, b^2 is also constant. S is the number of splices.

As the number of splices doubles, the time also appears to double. The one runtime that is not consistent with this pattern is the

last run because 4384 is a lot more than double of 109.968.

What patterns, if any, do you find in the runtimes for StringStrand?
 After tripling the StringStrand, the runtimes were as follows:

dna length = 13,917,663
 cutting at enzyme gaattc

Class	splicee	recomb	time	appends
StringStrand:	256	14,401,413	3.884	3870
StringStrand:	512	14,896,773	4.067	3870
StringStrand:	1,024	15,887,493	4.411	3870
StringStrand:	2,048	17,868,933	5.252	3870
StringStrand:	4,096	21,831,813	6.665	3870
StringStrand:	8,192	29,757,573	9.070	3870
StringStrand:	16,384	45,609,093	14.511	3870
StringStrand:	32,768	77,312,133	25.421	3870
StringStrand:	65,536	140,718,213	50.220	3870
StringStrand:	131,072	267,530,373	100.057	3870

Although there are indications of a doubling time, or time that follows $O(S)$, towards the end of the data, there is not a definite pattern seen in the runtimes overall. From time 3.884 to time 9.070 the increase is more gradual than the increase in splices.

2. Are the benchmark timings for StringBuilderStrand consistent with the explanation below that the time to execute cutAndSplice is $O(bS)$?

The runtimes of StringBuilderStrand are much faster than those of StringStrand, so it is a little more difficult to see an overall pattern.

However, from time 0.118 to time 0.558 S is large enough to see an increasing trend that follows $O(s)$. As with the earlier question, b is held constant, so $O(bs)$ is just $O(s)$.

What patterns, if any, do you find in the runtimes for StringBuilderStrand?
 After tripling the StringBuilderStrand, the runtimes were as follows:

dna length = 13,917,663
 cutting at enzyme gaattc

Class	splicee	recomb	time	appends
StringBuilderStrand:	256	14,401,413	0.018	3870
StringBuilderStrand:	512	14,896,773	0.018	3870
StringBuilderStrand:	1,024	15,887,493	0.018	3870
StringBuilderStrand:	2,048	17,868,933	0.025	3870
StringBuilderStrand:	4,096	21,831,813	0.021	3870
StringBuilderStrand:	8,192	29,757,573	0.033	3870
StringBuilderStrand:	16,384	45,609,093	0.060	3870
StringBuilderStrand:	32,768	77,312,133	0.050	3870
StringBuilderStrand:	65,536	140,718,213	0.056	3870
StringBuilderStrand:	131,072	267,530,373	0.165	3870
StringBuilderStrand:	262,144	521,154,693	0.271	3870
StringBuilderStrand:	524,288	1,028,403,333	0.507	3870

Here once again, there is not a clear pattern of the time doubling as the splices double. The first few runtimes are all about the same.

However, the last three runs again we begin to see a more steady increase in runtime with the larger splices.