

1. 插入前 1000, 5000, 10000, 25000, 50000 个数据的时间开销

将每个数据量下的插入操作运行十次，得数据如下表：

顺序表插入int数据	1000	5000	10000	25000	50000
1	2.28882	6.23384	11.6336	30.297	59.7571
2	1.06995	5.80882	13.4929	26.0012	53.8473
3	1.18072	5.03712	10.2712	26.1846	50.2117
4	2.13579	6.02584	10.1021	26.1912	58.4918
5	1.14667	5.61271	9.75425	29.074	50.1206
6	2.42913	4.47795	11.8039	25.6065	51.5585
7	1.6681	8.10707	12.1009	29.6	58.5374
8	1.36287	5.70912	11.865	27.6041	56.0648
9	3.17251	5.75959	11.5389	27.5832	62.5255
10	1.35097	6.93005	13.3116	27.9786	57.2709
平均值	1.780553	5.970211	11.58744	27.61204	55.83856

单链表插入int数据	1000	5000	10000	25000	50000
1	3.04	78.7404	255.792	1737.51	6727.72
2	2.75897	59.7062	205.703	1267.53	5024.04
3	2.98215	54.7995	193.401	1246.91	4861.71
4	2.44061	59.5208	217.997	1459.71	6086.78
5	2.49025	48.5464	188.3	1200.14	4839.95
6	2.42297	58.786	197.375	1246.04	4993.5
7	3.4642	58.779	232.566	1376.5	5572.22
8	3.13969	59.8822	228.246	1453.64	5527.96
9	4.0882	59.6689	226.472	1541.25	5568.74
10	3.22461	60.6896	241.721	1347.6	5368.09
平均值	3.005165	59.9119	218.7573	1387.683	5457.071

双链表插入int数据	1000	5000	10000	25000	50000
1	3.36	74.4873	289.014	1899.49	7564.6
2	2.43036	60.7835	222.911	1469.03	6745.65
3	2.47343	54.4652	222.61	1421.11	6113.17
4	3.8716	56.4562	201.242	1259.72	5051.78
5	2.42831	54.898	217.744	1415.47	6119.7
6	2.45169	59.9204	225.476	1416.27	5867.24
7	2.45374	61.9839	263.48	1641.17	6303.51
8	2.71795	55.403	239.615	1711.54	6569.14
9	2.78236	80.0684	264.436	1510.27	6229.53
10	2.92923	57.4707	249.739	1560.87	7323.1
平均值	2.789867	61.59366	239.6267	1530.494	6388.742

封装vector插入int数据	1000	5000	10000	25000	50000
1	6.90871	146.035	608.548	3732.49	16071.7
2	5.17497	118.681	462.375	2985.14	11110.8
3	5.41128	116.72	448.923	2754.61	10985.2
4	6.0123	121.272	480.446	2809.7	11451.1
5	5.43548	117.34	458.432	2886.6	11064.1
6	7.4133	122.207	455.28	2808.76	11171.7
7	5.61732	124.825	488.29	3204.98	12981.9
8	5.79282	149.882	503.602	3150.71	12317.6
9	5.1963	121.037	519.328	3096.71	12369.2
10	6.0681	138.774	491.223	3339.45	12411.7
平均值	5.903058	127.6773	491.6447	3076.915	12193.5

顺序表插入double数据	1000	5000	10000	25000	50000
1	11.1721	32.8607	75.6366	191.845	324.243
2	6.57026	26.7126	55.3075	131.257	344.045
3	6.7719	27.328	60.0526	137.335	346.665
4	6.99406	27.415	57.5927	169.422	358.008
5	11.1545	41.7485	58.2835	156.203	365.885
6	11.0622	33.5623	86.7373	188.937	333.836
7	7.85272	33.6944	60.6138	137.485	370.859
8	6.50831	30.7972	58.88	136.9	357.533
9	6.62442	29.5652	60.1473	165.052	324.125
10	8.42749	32.6897	58.2175	141.769	323.903
平均值	8.313796	31.63736	63.14688	155.6205	344.9102

单链表插入double数据	1000	5000	10000	25000	50000
1	6.60103	80.6782	340.762	2176.26	8833.87
2	6.81765	88.9687	365.17	2066.6	7756.34
3	10.0213	92.0497	346.267	1921.98	7676.25
4	7.25252	89.0602	357.154	2056.37	8270.61
5	7.09006	106.703	408.157	2082.92	9891.09
6	6.73436	82.8017	377.06	2171.99	7853.62
7	8.81888	95.9336	371.807	2108.83	7936.42
8	7.21642	88.9785	381.824	1996.19	8323.4
9	7.09047	88.5761	321.519	1956.04	8950.84
10	10.2129	138.136	463.783	2901.55	10882.3
平均值	7.785559	95.18857	373.3503	2143.873	8637.474

双链表插入double数据	1000	5000	10000	25000	50000
1	7.83631	102.252	418.454	2366.46	8811.5
2	9.38011	119.183	384.735	2154.17	8212.37
3	6.66011	85.0737	339.809	2117.64	8195.94
4	8.55878	92.4005	334.874	2013.66	8634.73
5	11.6862	134.371	471.979	2614.51	10598.5
6	6.50257	109.9	335.853	2100.38	8181.08
7	6.79016	110.604	356.218	2117.41	8083.35
8	7.10195	82.6315	328.413	2242.13	8280.18
9	11.8162	143.813	472.443	2861.04	10853.9
10	7.33006	89.5115	344.021	2146.64	9666.46
平均值	8.366245	106.974	378.6799	2273.404	8951.801

封装vector插入double数据		1000	5000	10000	25000	50000
1	10.5666	151.941	866.262	4885.08	17479.9	
2	9.93519	210.588	749.135	4154.53	16544.5	
3	10.8591	207.784	718.104	3937.57	16013.6	
4	10.5629	193.866	649.605	4005.11	17238	
5	16.4993	255.105	898.389	4127.35	15743.9	
6	15.7461	235.034	651.87	4209.58	16355.9	
7	10.1932	207.643	698.972	4585.8	17729.8	
8	20.9145	235.987	963.202	3865.95	20127.4	
9	14.8484	254.573	931.687	5467.93	21344.4	
10	16.6252	337.774	926.401	5449.11	21340.3	
平均值	13.675049	229.0295	805.3627	4468.801	17991.77	

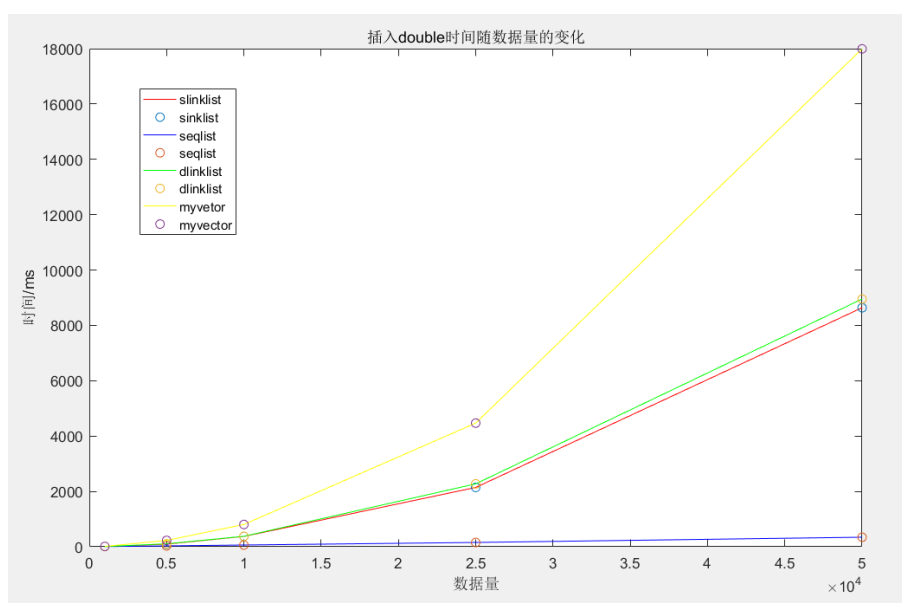
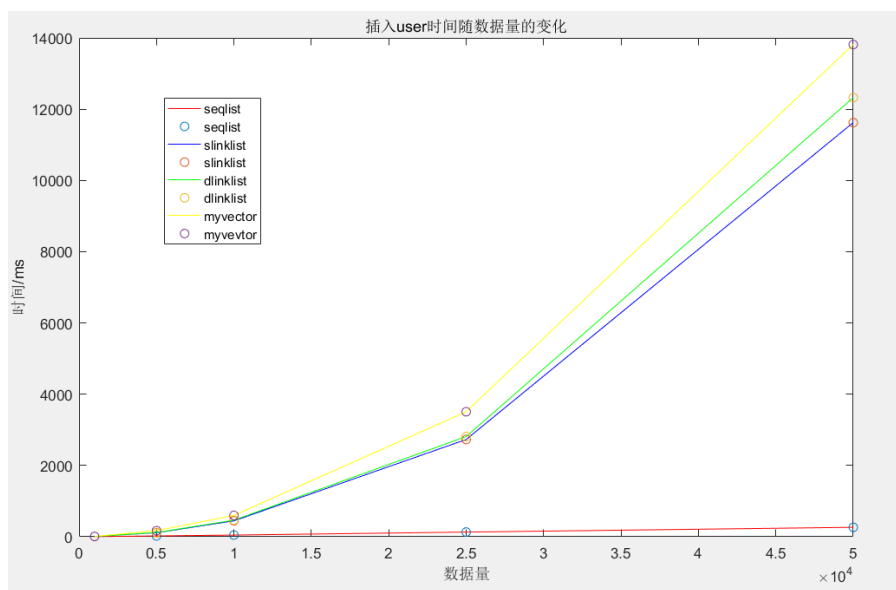
顺序表插入user数据		1000	5000	10000	25000	50000
1	4.38031	25.0371	45.2251	124.011	239.846	
2	4.3717	20.9375	43.1594	117.862	244.839	
3	4.19405	21.9734	45.0708	129.188	245.769	
4	4.23959	21.6279	45.8778	122.067	239.369	
5	5.20985	21.559	45.502	127.394	241.507	
6	4.36267	22.4095	45.2164	123.257	239.221	
7	5.56062	22.8808	48.1547	131.582	249.808	
8	5.41375	24.9239	52.757	142.326	361.626	
9	5.24431	23.9155	51.3908	138.565	291.175	
10	5.63734	24.5465	50.2667	148.49	294.479	
平均值	4.861419	22.98111	47.26207	130.4742	264.7639	

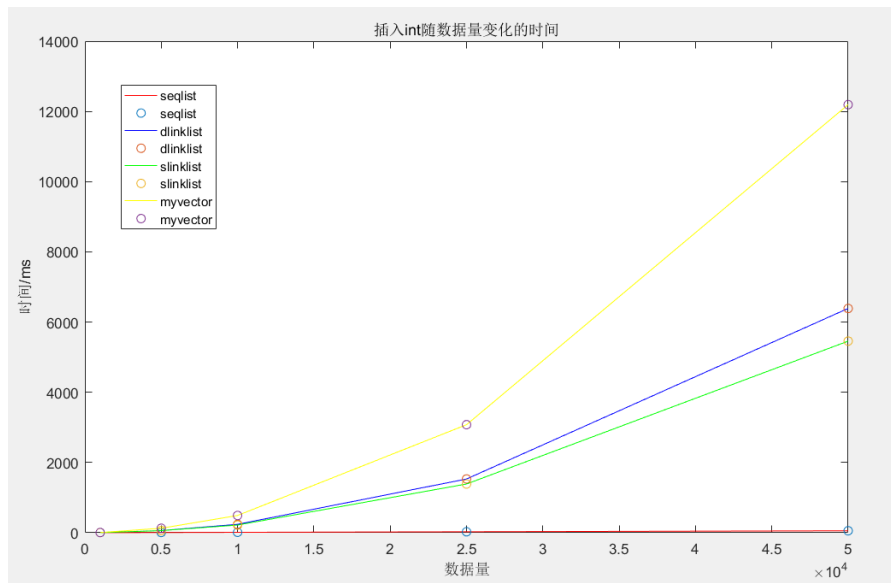
单链表插入user数据		1000	5000	10000	25000	50000
1	7.69108	105.64	403.501	2425.49	9866.64	
2	6.45867	105.235	401.846	2613.58	10342.1	
3	7.11754	103.571	410.403	2420.31	9806.94	
4	6.07221	108.048	418.987	2490.45	9864.26	
5	6.44062	114.627	410.191	2610.01	9979.82	
6	6.42011	102.098	423.843	2700.67	10331.9	
7	7.78667	105.872	413.853	2490.19	10543.5	
8	9.15611	130.785	572.834	3830.31	20448	
9	7.08883	113.836	473.316	2769.87	12209.4	
10	7.75098	130.101	531.67	2939.86	12846.6	
平均值	7.198282	111.9813	446.0444	2729.074	11623.92	

双链表插入user数据		1000	5000	10000	25000	50000
1	7.73498	104.625	390.18	2363.19	10881.1	
2	6.20144	100.99	427.382	2471.55	10347.7	
3	7.86257	105.149	406.194	2425.24	10663.4	
4	6.19857	100.763	404.7	2463.31	10373.9	
5	6.36226	108.034	406.302	2525.22	10256.4	
6	7.60083	108.279	394.244	2483.07	10205.2	
7	7.80308	110.219	438.021	2539.03	10523.3	
8	10.8599	193.647	748.178	4409.35	20595.4	
9	9.01703	129.142	568.303	3372.7	13865.7	
10	9.77273	125.095	459.838	3057.2	15519	
平均值	7.941339	118.5943	464.3342	2810.986	12323.11	

封装vector插入user数据	1000	5000	10000	25000	50000
1	9.8396	138.506	497.301	3045.77	11764.8
2	8.97683	211.792	498.982	3007.11	11967.3
3	8.80329	139.687	506.451	3005.46	11714.6
4	8.72803	134.885	506.21	3070.13	11857.8
5	8.70319	135.577	514.043	3022.64	11758.4
6	8.45129	138.696	507.836	2987.04	11847.9
7	10.2318	232.027	747.909	3996.78	16280.6
8	16.6051	249.989	927.064	5397.72	21459
9	9.75221	163.39	620.413	3672.14	14853.5
10	10.0927	168.296	658.862	3884.44	14626.4
平均值	10.018404	171.2845	598.5071	3508.923	13813.03

对表中数据分别取平均值后，利用 matlab 进行绘图，得图像如下所示





分析：从图像中可以看出，不管哪种类型的对象，所需时间都是顺序表<单链表<双链表<封装 vector，其中单链表和双链表所需的时间十分接近。顺序表插入所需时间与数据量近似于线性关系，因此顺序表插入时间随数据量的增长并不明显，而其他三种实现方式随数据量的增长，插入所需时间大幅增长。

书本上说顺序表的插入操作性能较差，而链表的插入操作性能较好，但这是相对于随机插入而言的，此处的情况是依次插入，因此顺序表不存在为了插入一个元素而将其后所有元素后移的情况，而链表需要多次移动指针，所以顺序表时间性能在此种情况下反而优于链表（我是用迭代器对 vector 进行插入操作，因此所需时间相当长，猜测用下标访问会节省大量时间）。至于单链表、双链表和封装 vector 时间性能上出现差异的可能原因个人猜测为单链表、双链表、封装 vector 开辟的空间依次变大，导致所需时间依次变大。

但是在实际应用中会出现很多在中间插入的情况，因此就实际应用来说链表在插入操作上的性能还是优于顺序表。

2. 在数据集数量为 50000 的情况下，随机生成 1000 个访问下标，测量访问 1000 个元素的时间开销

访问1000个int数的时间	seqlist	slinklist	dlinklist	myvector
1	0.0406154	2.86893	3.18605	0.043077
2	0.0254359	1.88718	2.87016	0.059875
3	0.0254359	1.36862	1.6	0.060718
4	0.0295385	2.54195	2.54072	0.059897
5	0.0365128	2.63056	2.44923	0.057026
平均值ms	0.0315077	2.259448	2.529232	0.056119
访问单个数据平均时间μs	0.0315077	2.259448	2.529232	0.056119

访问1000个double数的时间	seqlist	slinklist	dlinklist	myvector
1	0.0278975	2.75857	2.65682	0.06318
2	0.0418462	2.8238	3.00513	0.072205
3	0.0221538	2.67979	1.34646	0.062769
4	0.0233846	2.67282	1.59754	0.060308
5	0.0332308	2.64697	1.23856	0.06318
平均值ms	0.02970258	2.71639	1.968902	0.064328
访问单个数据平均时间 μ s	0.02970258	2.71639	1.968902	0.064328

访问1000个user的时间	seqlist	slinklist	dlinklist	myvector
1	0.523488	3.27344	4.28841	0.349949
2	0.59118	4.23754	2.78769	0.316718
3	0.523487	4.17395	2.2199	0.549744
4	0.524718	2.43364	4.9231	0.500103
5	0.523897	3.88513	3.0199	0.353231
平均值ms	0.537354	3.60074	3.4478	0.413949
访问单个数据平均时间 μ s	0.537354	3.60074	3.4478	0.413949

(*注: 封装 vector 的 visit 操作由下标访问实现)

不同数据类型下中随机访问 1000 个元素所需的时间如上表所示, 每个数据集进行了 5 次试验, 对结果求取平均值, 并得到访问单个元素所需平均时间。

由表中数据可以看出不论哪种数据类型, 顺序表和封装 vector 的实现方式所需时间都比链表实现方式小了一个数量级, 所以可以得出对于访问操作顺序表的性能优于链表, 原因在于通过下标访问无需多次移动指针, 所以采取下标访问的封装 vector 时间性能与顺序表在同一数量级。

3. 在数据集数量为 50000 的情况下, 从提供的数据集中随机选择 1000 个元素, 测量删除这 1000 个元素的时间开销

删除1000个int的时间	seqlist	slinklist	dlinklist	myvector
1	156.783	1.44944	1.55241	10.0197
2	176.071	1.43097	1.43179	8.06072
3	190.866	1.08226	1.59426	7.94585
4	178.294	1.31241	1.51221	11.5766
5	204.796	1.51549	1.31282	12.4665
平均值ms	181.362	1.358114	1.480698	10.01387
删除单个元素平均时间 μ s	181.362	1.358114	1.480698	10.01387

删除1000个double的时间	seqlist	slinklist	dlinklist	myvector
1	209.868	0.989538	1.47528	27.1106
2	217.975	1.50031	1.48308	22.1555
3	228.013	2.36759	1.40513	23.3616
4	226.958	1.42851	1.53436	28.3692
5	228.215	1.54995	0.969846	27.0055
平均值ms	222.2058	1.56718	1.373539	25.60048
删除单个元素平均时间 μ s	222.2058	1.56718	1.373539	25.60048

删除1000个user的时间		seqlist	slinklist	dlinklist	myvector
	1	3263.58	2.31138	2.28308	3216.91
	2	2944.85	2.17723	2.22605	2685.45
	3	3009.42	1.97046	1.51508	2901.2
	4	3110.52	1.51426	2.49887	3161.92
	5	3076.81	2.34092	2.24944	3182.66
平均值ms		3081.036	2.06285	2.154504	3029.628
删除单个元素平均时间 μ s		3081.036	2.06285	2.154504	3029.628

不同数据类型下中随机删除 1000 个元素所需的时间如上表所示，每个数据集进行了 5 次试验，对结果求取平均值，并得到访问单个元素所需平均时间。

由表中数据可以看出不论哪种数据类型，单链表和双链表执行删除操作所需的时间都远小于另外两种实现方式，因为对于链表删除后无需依次向前移动后面的元素，所以对于删除操作链表优于顺序表。

4. 书本程序题见源代码