

Implementation	Conclusion																																								
<p>List</p> <table> <tr> <td colspan="2">Python list library</td></tr> <tr> <td>Operation</td><td>Timing result</td></tr> <tr> <td>Front insert</td><td>6.17967605591e-07</td></tr> <tr> <td>Back insert</td><td>1.2829208374e-07</td></tr> <tr> <td>TRAVERSAL</td><td>6.55040740967e-08</td></tr> <tr> <td>Front delete</td><td>2.19036102295e-07</td></tr> <tr> <td>Back delete</td><td>9.74836349487e-08</td></tr> </table> <table> <tr> <td colspan="2">Arraylist implementation</td></tr> <tr> <td>Operation</td><td>Timing result</td></tr> <tr> <td>Front insert</td><td>7.01058769226e-06</td></tr> <tr> <td>Back insert</td><td>8.1794834137e-07</td></tr> <tr> <td>TRAVERSAL</td><td>8.09164047241e-07</td></tr> <tr> <td>Front delete</td><td>4.73778057098e-06</td></tr> <tr> <td>Back delete</td><td>1.46316814423e-06</td></tr> </table> <table> <tr> <td colspan="2">Pointerlist implementation</td></tr> <tr> <td>Operation</td><td>Timing result</td></tr> <tr> <td>Front insert</td><td>2.72169494629e-06</td></tr> <tr> <td>Back insert</td><td>0.000261447299004</td></tr> <tr> <td>TRAVERSAL</td><td>1.35574817657e-06</td></tr> <tr> <td>Front delete</td><td>7.17076301575e-07</td></tr> </table>	Python list library		Operation	Timing result	Front insert	6.17967605591e-07	Back insert	1.2829208374e-07	TRAVERSAL	6.55040740967e-08	Front delete	2.19036102295e-07	Back delete	9.74836349487e-08	Arraylist implementation		Operation	Timing result	Front insert	7.01058769226e-06	Back insert	8.1794834137e-07	TRAVERSAL	8.09164047241e-07	Front delete	4.73778057098e-06	Back delete	1.46316814423e-06	Pointerlist implementation		Operation	Timing result	Front insert	2.72169494629e-06	Back insert	0.000261447299004	TRAVERSAL	1.35574817657e-06	Front delete	7.17076301575e-07	<p>The input size is 500. The python list library is obviously faster than the two implementation made by me. And all five operations is faster in arraylist than in pointerlist. The back insert and back delete takes really long time to compile in pointerlist.</p>
Python list library																																									
Operation	Timing result																																								
Front insert	6.17967605591e-07																																								
Back insert	1.2829208374e-07																																								
TRAVERSAL	6.55040740967e-08																																								
Front delete	2.19036102295e-07																																								
Back delete	9.74836349487e-08																																								
Arraylist implementation																																									
Operation	Timing result																																								
Front insert	7.01058769226e-06																																								
Back insert	8.1794834137e-07																																								
TRAVERSAL	8.09164047241e-07																																								
Front delete	4.73778057098e-06																																								
Back delete	1.46316814423e-06																																								
Pointerlist implementation																																									
Operation	Timing result																																								
Front insert	2.72169494629e-06																																								
Back insert	0.000261447299004																																								
TRAVERSAL	1.35574817657e-06																																								
Front delete	7.17076301575e-07																																								

<table> <tr> <td>Back delete</td><td>0.000786135935783</td></tr> </table>	Back delete	0.000786135935783																	
Back delete	0.000786135935783																		
<p>Stack</p> <p>Python stack library</p> <table> <tr> <th>Operation</th><th>Timing result</th></tr> <tr> <td>PUSH</td><td>1.27815246582e-07</td></tr> <tr> <td>POP</td><td>2.88806915283e-07</td></tr> </table> <p>Arraystack</p> <table> <tr> <th>Operation</th><th>Timing result</th></tr> <tr> <td>PUSH</td><td>4.35056686401e-07</td></tr> <tr> <td>POP</td><td>7.32172012329e-07</td></tr> </table> <p>Pointerstack</p> <table> <tr> <th>Operation</th><th>Timing result</th></tr> <tr> <td>PUSH</td><td>1.04679203033e-06</td></tr> <tr> <td>POP</td><td>1.36152839661e-06</td></tr> </table>	Operation	Timing result	PUSH	1.27815246582e-07	POP	2.88806915283e-07	Operation	Timing result	PUSH	4.35056686401e-07	POP	7.32172012329e-07	Operation	Timing result	PUSH	1.04679203033e-06	POP	1.36152839661e-06	<p>The input size is 500. The python stack library is obviously faster than the implementation made by me. The the operation PUSH and POP is both faster in arraystack than in pointerstack.</p>
Operation	Timing result																		
PUSH	1.27815246582e-07																		
POP	2.88806915283e-07																		
Operation	Timing result																		
PUSH	4.35056686401e-07																		
POP	7.32172012329e-07																		
Operation	Timing result																		
PUSH	1.04679203033e-06																		
POP	1.36152839661e-06																		