

Modul 4 – *Linked List* dan *Sorting*

Praktikum Struktur Data – 2019

1. Buatlah class `LinkedList`, dengan beberapa method tambahan pada class `LinkedList` seperti berikut (untuk constructor `LinkedList`, dan class `Node` dapat dilihat pada materi perkuliahan, serta method `__str__` sama seperti praktikum sebelumnya):
 - a. `insertPrevious` : yang digunakan untuk menyisipkan sebuah node sebelum node tertentu
 - b. `insertNext` : yang digunakan untuk menyisipkan sebuah node sesudah node tertentu

Berikut adalah contoh penggunaan kedua method tersebut.

Perintah `insertNext(84,100)` adalah menyisipkan node dengan data=100, sesudah node dengan data = 84.

Sedangkan perintah `insertPrevious(5,33)` adalah menyisipkan node dengan data 33 sebelum node dengan data = 5.

```
In [3]: mylist1=LinkedList()
mylist1.addRear(5)
mylist1.addRear(84)
mylist1.addRear(12)
mylist1.addRear(77)
print(mylist1)
[5,84,12,77]

In [4]: mylist1.insertNext(84,100)
print(mylist1)
[5,84,100,12,77]

In [5]: mylist1.insertPrevious(5,33)
print(mylist1)
[33,5,84,100,12,77]
```

2. Buatlah class `Sorting`, dengan property/state adalah data yang akan diurutkan (berbentuk list), dengan beberapa method berikut ini :
 - a. Constructor : untuk inisialisasi property/state dari object
 - b. `bubbleSort`, `selectionSort`, dan `insertionSort` :
sorting dengan menggunakan algoritma bubble sort, atau selection sort, atau insertion sort, dimana terdapat parameter berupa jenis sorting, yaitu 'a' berarti hasil sorting adalah **ascending**, sedangkan 'd' berarti hasil sorting adalah **descending**. Data yang sudah diurutkan disimpan dalam property/state object tersebut. Return value, berupa waktu komputasi yang dibutuhkan untuk eksekusi algoritma ini

Petunjuk :

- data yang akan diurutkan adalah data yang degenerate secara random, gunakan modul random
- untuk menghitung waktu eksekusi, gunakan modul time, dan syntax time.time()

Berikut adalah contoh penggunaan class Sorting dan eksekusi method salah satu algoritma sorting.

```
In [2]: import random
a=random.sample(range(0, 1000), 500)
dataSort=Sorting(a)
t1=dataSort.bubbleSort('a')
print(dataSort.data)
print('waktu komputasi = ', t1, 'detik')
```

[5, 6, 11, 12, 14, 17, 19, 24, 27, 28, 30, 31, 33, 34, 35, 39, 43, 48, 49, 57, 58, 59, 60, 63, 66, 69, 72, 73, 78, 83, 84, 85, 86, 87, 89, 91, 92, 98, 99, 100, 101, 102, 103, 105, 107, 108, 110, 111, 115, 118, 119, 125, 126, 128, 129, 131, 132, 134, 138, 143, 145, 146, 147, 148, 149, 152, 153, 155, 156, 159, 160, 161, 163, 164, 165, 166, 168, 169, 172, 173, 174, 175, 176, 178, 182, 183, 184, 186, 188, 193, 195, 196, 197, 199, 200, 201, 202, 204, 206, 210, 215, 218, 224, 225, 226, 233, 242, 244, 247, 250, 251, 252, 255, 256, 262, 263, 264, 265, 266, 267, 268, 269, 271, 272, 275, 277, 279, 280, 281, 284, 286, 287, 288, 290, 293, 294, 296, 297, 298, 299, 300, 301, 303, 304, 305, 307, 308, 309, 310, 311, 313, 314, 315, 318, 320, 321, 324, 326, 328, 330, 331, 332, 333, 334, 336, 342, 343, 344, 346, 347, 348, 349, 350, 352, 358, 361, 363, 365, 367, 368, 369, 371, 375, 376, 379, 380, 381, 384, 385, 386, 391, 395, 396, 399, 400, 403, 404, 406, 409, 412, 414, 418, 419, 420, 427, 429, 431, 432, 433, 434, 436, 437, 438, 440, 441, 443, 444, 445, 446, 448, 450, 453, 454, 455, 457, 458, 462, 463, 464, 465, 468, 473, 476, 477, 479, 481, 484, 485, 487, 491, 493, 494, 495, 496, 499, 500, 502, 503, 504, 508, 509, 510, 513, 515, 520, 527, 528, 529, 531, 532, 534, 535, 538, 544, 545, 546, 547, 548, 550, 551, 553, 556, 558, 563, 564, 566, 567, 568, 571, 574, 576, 577, 580, 581, 583, 586, 588, 589, 590, 591, 594, 595, 600, 601, 603, 604, 607, 608, 611, 612, 613, 614, 615, 616, 617, 618, 619, 621, 622, 625, 629, 630, 632, 633, 634, 637, 640, 645, 647, 648, 649, 652, 654, 655, 656, 661, 663, 667, 670, 674, 675, 676, 680, 682, 683, 686, 688, 690, 691, 693, 695, 696, 698, 700, 702, 704, 705, 708, 713, 714, 715, 716, 717, 718, 721, 723, 730, 733, 734, 737, 738, 740, 743, 744, 748, 750, 753, 758, 759, 761, 764, 765, 767, 772, 773, 774, 775, 776, 778, 779, 782, 785, 787, 788, 789, 791, 792, 793, 794, 797, 800, 802, 803, 804, 805, 808, 809, 811, 812, 813, 814, 815, 816, 818, 819, 820, 821, 822, 824, 825, 827, 829, 831, 833, 835, 836, 840, 843, 845, 848, 849, 852, 854, 855, 864, 866, 870, 871, 873, 874, 875, 876, 877, 880, 882, 883, 884, 886, 888, 889, 890, 891, 892, 893, 895, 896, 898, 899, 902, 903, 904, 905, 908, 910, 911, 912, 914, 915, 918, 919, 921, 922, 923, 924, 934, 935, 939, 940, 942, 944, 945, 946, 947, 949, 950, 954, 955, 957, 958, 959, 960, 961, 963, 964, 965, 966, 967, 970, 971, 974, 979, 983, 985, 986, 987, 988, 991, 995, 997, 998]

waktu komputasi = 0.1668086051940918 detik

```
In [3]: t1=dataSort.insertionSort('d')
print(dataSort.data)
print('waktu komputasi = ', t1, 'detik')
```

[998, 997, 995, 991, 988, 987, 986, 985, 983, 979, 974, 971, 970, 967, 966, 965, 964, 963, 961, 960, 959, 958, 957, 955, 954, 950, 949, 947, 946, 945, 944, 942, 940, 939, 935, 934, 924, 923, 922, 921, 919, 918, 915, 914, 912, 911, 910, 908, 905, 904, 903, 902, 899, 898, 896, 895, 893, 892, 891, 890, 889, 888, 886, 884, 883, 882, 880, 877, 876, 875, 874, 873, 871, 870, 866, 864, 855, 854, 852, 849, 848, 845, 843, 840, 836, 835, 833, 831, 829, 827, 825, 824, 822, 821, 820, 819, 818, 816, 815, 814, 813, 812, 811, 809, 808, 805, 804, 803, 802, 800, 797, 794, 793, 792, 791, 789, 788, 787, 785, 782, 779, 778, 776, 775, 774, 773, 772, 767, 765, 764, 761, 759, 758, 753, 750, 748, 744, 743, 740, 738, 737, 734, 733, 730, 723, 721, 718, 717, 716, 715, 714, 713, 708, 705, 704, 702, 700, 698, 696, 695, 693, 691, 690, 688, 686, 683, 682, 680, 676, 675, 674, 670, 667, 663, 661, 656, 655, 654, 652, 649, 648, 647, 645, 640, 637, 636, 633, 632, 630, 629, 625, 622, 621, 619, 618, 617, 616, 615, 614, 613, 612, 611, 608, 607, 604, 603, 601, 600, 595, 594, 591, 590, 589, 588, 586, 583, 581, 580, 577, 576, 574, 571, 568, 567, 566, 564, 563, 558, 556, 553, 551, 550, 548, 547, 546, 545, 544, 538, 535, 534, 532, 531, 529, 528, 527, 520, 515, 513, 510, 509, 508, 504, 503, 502, 500, 499, 496, 495, 494, 493, 491, 487, 485, 484, 481, 479, 477, 476, 473, 468, 465, 464, 463, 462, 458, 457, 455, 454, 453, 450, 448, 446, 445, 444, 443, 441, 440, 438, 437, 436, 434, 433, 432, 431, 429, 427, 420, 419, 418, 414, 412, 409, 406, 404, 403, 400, 399, 396, 395, 391, 386, 385, 384, 381, 380, 379, 376, 375, 371, 369, 368, 367, 365, 363, 361, 358, 352, 350, 349, 348, 347, 346, 344, 343, 342, 336, 334, 333, 332, 331, 330, 328, 326, 324, 321, 320, 318, 315, 314, 313, 311, 310, 309, 308, 307, 305, 304, 303, 301, 300, 299, 298, 297, 296, 294, 293, 290, 288, 287, 286, 284, 281, 280, 279, 277, 275, 272, 271, 269, 268, 267, 266, 265, 264, 263, 262, 256, 255, 252, 251, 250, 247, 244, 242, 233, 226, 225, 224, 218, 215, 210, 206, 204, 202, 201, 200, 199, 197, 196, 195, 193, 188, 186, 184, 183, 182, 178, 176, 175, 174, 173, 172, 169, 168, 166, 165, 164, 163, 161, 160, 159, 156, 155, 154, 152, 149, 148, 147, 146, 145, 143, 138, 134, 132, 131, 129, 128, 126, 125, 119, 118, 115, 111, 110, 108, 107, 105, 103, 102, 101, 100, 99, 98, 92, 91, 89, 87, 86, 85, 84, 83, 78, 73, 72, 69, 66, 63, 60, 59, 58, 57, 49, 48, 43, 39, 35, 34, 33, 31, 30, 28, 27, 24, 19, 17, 14, 12, 11, 6, 5]

waktu komputasi = 0.11970329284667969 detik