

A doubly linked list – Self Testing

Test your functionalities properly and seriously.

Mark Pass or Fail for each operation. If your implementation and timing should work properly and correctly. If your timing does not match with your code or does not work, you will not get a full credit or even get a penalty for your implementation.

Step	Operations	Point	Testing	comments
1	find, more, less	0.5	PASS	Check the code. find() use one while loop, but not if Use push command to test find().
2	push commands push()	0.5	PASS	Check the code. push() must use find() and insert(), not more than 3~4 lines of code Don't add a new node if the position x is not found
3	pop commands pop_all*	0.5	PASS	Test it with over 100,000 samples. Make sure O(n), not O(n^2) Use testing method described below
4	half() and show()*	0.5	PASS	Check the correctness the middle node. half() is used in shuffle() and show() Test it with both odd & even number sequences. Record the timing of half() for 20 million samples displayed using show HEAD/TAIL on exiting. Method 1: 0.066205 sec Method 2: 0.048796 sec Method 2 is faster than Method 1 by ____26.3____ %
5	swap_pairs	0.0	PASS	Check the code. It must go through the list once, not twice nor more. Test it with both odd & even number sequences.
6	sorted()	0.5	PASS	It is checked by other operations.
7	push_sorted()	0.5	PASS	Check it with unsorted, ascending and descending ordered lists. Make sure duplicated ones included such as 3 5 5 7 9 9 9. Use "reverse" menu option. Test it with over 100,000 samples. pushN to generate samples and push_sorted 100001. Make sure O(n), not O(n^2) Additionally, use testing method described below
8	unique()*	0.5	PASS	Test it with over 100,000 samples. Make sure O(n), not O(n^2) Use testing method described below
9	reverse()	0.0	PASS	Test it with over 100,000 samples. Make sure O(n), not O(n^2)
10	randomize()	0.0	PASS	Test it with over 100,000 samples. Make sure O(n), not O(n^2) The commands sort & quicksort uses randomize().

11	shuffle()*	0.5	PASS	Check the exactness. Test it with both odd & even number sequences.
	Total	4.0	0.8 + 0.2	Extra 0.1 p per step for a proper testing Extra 0.2 if you get them all right

Test Hint 1: pop_all()

To test pop_all(), you may need to generate a sequence that has a consecutive numbers of a certain value. You may use "push back N" command option with a negative N provided.

For example, make a sequence with ten thousands and another ten thousands of 7 samples:

- select "push_back_N" and enter 10,000 for random samples
- select "push_back_N" and enter -10000, then enter "7" for a value.
- run "pop-all" 7.

Test Hint 2: unique()

To test unique(), you may also need to generate a sorted sequence with consecutive numbers of a certain value. You may use "push back N" command option with a negative N provided.

For example, make a sequence with thirty thousands for each 1, 5, and 7, and ten thousands for 9, respectively and run unique().

Test Hint 3: show(), pop(), push_sorted(), push_backN(), ...

Make a sequence of numbers from 1 to 100 as shown below in a fewer steps possible. Then you may need to use all kinds of commands you implemented so far.

```

b - push back N      0(n)      r - reverse      0(n)
Y - pop back N      0(n)      x - shuffle**   0(n)
c - clear            0(n)      t - show [HEAD/TAIL]
                                n - show n nodes per line

Command[q to quit]: t
-> 51 -> 1 -> 52 -> 2 -> 53 -> 3 -> 54 -> 4 -> 55 -> 5
-> 56 -> 6 -> 57 -> 7 -> 58 -> 8 -> 59 -> 9 -> 60 -> 10
-> 61 -> 11 -> 62 -> 12 -> 63 -> 13 -> 64 -> 14 -> 65 -> 15
-> 66 -> 16 -> 67 -> 17 -> 68 -> 18 -> 69 -> 19 -> 70 -> 20
-> 71 -> 21 -> 72 -> 22 -> 73 -> 23 -> 74 -> 24 -> 75 -> 25
-> 76 -> 26 -> 77 -> 27 -> 78 -> 28 -> 79 -> 29 -> 80 -> 30
-> 81 -> 31 -> 82 -> 32 -> 83 -> 33 -> 84 -> 34 -> 85 -> 35
-> 86 -> 36 -> 87 -> 37 -> 88 -> 38 -> 89 -> 39 -> 90 -> 40
-> 91 -> 41 -> 92 -> 42 -> 93 -> 43 -> 94 -> 44 -> 95 -> 45
-> 96 -> 46 -> 97 -> 47 -> 98 -> 48 -> 99 -> 49 -> 100 -> 50

Doubly Linked List( 100 nodes, 10 nodes per line)
f - push front      0(1)      p - pop front   0(1)
b - push back       0(1)      y - pop back    0(1)

```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadoe8643ffbea667e6c

Step1. Find, more, less

Find :

```
Enter a number to push back( N for a value): 1
cpu: 3.9e-05 sec
FRONT 3 1 7 3 2 0 5

Doubly Linked List(nodes:7, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: i
Enter a number to push: 6
Choose a position node: 7
FRONT 3 1 6 7 3 2 0 5

Doubly Linked List(nodes:8, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
```

More :

```
Enter a number to push: 9
cpu: 3.1e-05 sec
FRONT 3 5 7 9

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: z
Enter a number to push: 6
cpu: 3.9e-05 sec
FRONT 3 5 6 7 9

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
```

Less :

```
Command[q to quit]: r
cpu: 3.3e-05 sec
FRONT 9 7 5 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: z
Enter a number to push: 6
cpu: 0.000329 sec
FRONT 9 7 6 5 3

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
```

Step2. Push commands

Push() :

```
cpu: 0.00014 sec
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: i
Enter a number to push: 2
Choose a position node: 0
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
```

```
Choose a position node: 0
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: i
Enter a number to push: 0
Choose a position node: 2
FRONT 1 0 2 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

Push_back() :

```
Choose a position node: 2
FRONT 1 0 2 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: b
Enter a number to push: 4
FRONT 1 0 2 3 4

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
```

Push_back(int N) :

```
cpu: 0.00014 sec
FRONT 1 0

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: B
Enter N nodes to push back(-N for a value)? : 2

cpu: 6.2e-05 sec
FRONT 1 0 4 2

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoad doe8643ffbea667e6c

Push_back(int N, int value) :

```
cpu: 0.000000 sec
FRONT 1 0 4 2

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back      0(1)  y - pop back  0(1)
B - push back N    0(n)  Y - pop back N 0(n)
i - push           0(n)  d - pop      0(n)
z - push sorted*   0(n)  e - pop all* 0(n)

s - sorted?        0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n) a - randomize  0(n)
u - unique*        0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear          0(n)
Command[q to quit]: B
Enter N nodes to push back(-N for a value)? : -3
Enter a value to push back?: 3

pushing [4]=3
cpu: 0.000112 sec
FRONT 1 0 4 2 3 3 3

Doubly Linked List(nodes:7, show:HEAD/TAIL,10)
```

Push_front() :

```
cpu: 2.4e-05 sec
FRONT 3 3 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back      0(1)  y - pop back  0(1)
B - push back N    0(n)  Y - pop back N 0(n)
i - push           0(n)  d - pop      0(n)
z - push sorted*   0(n)  e - pop all* 0(n)

s - sorted?        0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n) a - randomize  0(n)
u - unique*        0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear          0(n)
Command[q to quit]: f
Enter a number to push: 1
FRONT 1 3 3 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

Step3. Pop commands

Pop() :

```
cpu: 4.1e-05 sec
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back      0(1)  y - pop back  0(1)
B - push back N    0(n)  Y - pop back N 0(n)
i - push           0(n)  d - pop      0(n)
z - push sorted*   0(n)  e - pop all* 0(n)

s - sorted?        0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n) a - randomize  0(n)
u - unique*        0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear          0(n)
Command[q to quit]: d
Enter a number to pop: 2
cpu: 1.8e-05 sec
FRONT 1 3

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back      0(1)  y - pop back  0(1)
```

```
cpu: 3.8e-05 sec
FRONT 1 4 1 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back      0(1)  y - pop back  0(1)
B - push back N    0(n)  Y - pop back N 0(n)
i - push           0(n)  d - pop      0(n)
z - push sorted*   0(n)  e - pop all* 0(n)

s - sorted?        0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n) a - randomize  0(n)
u - unique*        0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear          0(n)
Command[q to quit]: d
Enter a number to pop: 2
cpu: 2.1e-05 sec
FRONT 1 4 1 3

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbca0add0e8643ffbea667e6c

Pop_front() :

```
cpu: 1.8e-05 sec
FRONT 1 3

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop    0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: p
cpu: 6e-05 sec
FRONT 3

Doubly Linked List(nodes:1, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
```

Pop_back() :

```
Enter a number to push: 4
FRONT 3 4

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop    0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: y
FRONT 3

Doubly Linked List(nodes:1, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
```

Pop_backN() :

```
Enter a number to push: 5
FRONT 3 1 4 2 5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop    0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: Y
Enter a number of nodes to pop back? :3

popping [4]
cpu: 3.9e-05 sec
FRONT 3 1

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

```
cpu: 3.9e-05 sec
FRONT 3 1

Doubly Linked List(nodes:2, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop    0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: Y
Enter a number of nodes to pop back? :0

popping [1]
cpu: 0.000114 sec

The list is empty.

Doubly Linked List(nodes:0, show:HEAD/TAIL,10)
```


// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadddoe8643ffbea667e6c

```
cpu: 4e-05 sec
FRONT 1 2 0 0

Doubly Linked List(nodes:4, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: Y
Enter a number of nodes to pop back? :10

popping [3]
cpu: 4.6e-05 sec

The list is empty.

Doubly Linked List(nodes:0, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
```

Pop_all() :

```
The list is empty.

Doubly Linked List(nodes:0, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: B
Enter N nodes to push back(-N for a value)? : 50000

cpu: 0.005411 sec
FRONT 44338 28912 27160 44027 48676 8837 46601 48091 37737 16823
... 5560 ...
24854 33993 28952 27273 13348 44403 5564 14628 8604 5052

Doubly Linked List(nodes:50000, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

```
cpu: 0.005411 sec
FRONT 44338 28912 27160 44027 48676 8837 46601 48091 37737 16823
... 5560 ...
24854 33993 28952 27273 13348 44403 5564 14628 8604 5052

Doubly Linked List(nodes:50000, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: B
Enter N nodes to push back(-N for a value)? : -50000
Enter a value to push back?: 7

pushing [50000]=7
pushing [60000]=7
pushing [70000]=7
pushing [80000]=7
pushing [90000]=7
cpu: 0.005036 sec
FRONT 44338 28912 27160 44027 48676 8837 46601 48091 37737 16823
... 7 ...
7 7 7 7 7 7 7 7 7 7

Doubly Linked List(nodes:100000, show:HEAD/TAIL,10)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadoe8643ffbea667e6c

```
cpu: 0.005036 sec
FRONT 44338 28912 27160 44027 48676 8837 46601 48091 37737 16823
... 7 ...
7 7 7 7 7 7 7 7

Doubly Linked List(nodes:100000, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: e
Enter a number to pop all: 7
cpu: 0.008731 sec
FRONT 44338 28912 27160 44027 48676 8837 46601 48091 37737 16823
... 5560 ...
24854 33993 28952 27273 13348 44403 5564 14628 8604 5052

Doubly Linked List(nodes:49998, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
```

Step4. Half() and show()*

Odd number :

```
cpu: 4.1e-05 sec
FRONT 1 2 3 4 5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: n
Enter number of nodes to show per line: 2
FRONT 1 2
... 3 ...
4 5

Doubly Linked List(nodes:5, show:HEAD/TAIL,2)
f - push front 0(1) p - pop front 0(1)
```

Even number :

```
Enter a number to push: 6
FRONT 1 2 3 4 5 6

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: n
Enter number of nodes to show per line: 2
FRONT 1 2
... 4 ...
5 6

Doubly Linked List(nodes:6, show:HEAD/TAIL,2)
f - push front 0(1) p - pop front 0(1)
```


// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadoe8643ffbea667e6c

Method 1 :

```
54 // For example, for list [0, 1, 2, 3, 4, 5, 6, 7], it returns 4.
55 // For example, for list [0, 1, 2, 3, 4, 5, 6, 7], it returns 4.
56 #if 1 // method 1 - slower version
57 pNode half(plist p) {
58     int N = size(p);
59     Node *curr = begin(p);
60     for (int i = 0; i < N / 2; i++)
61         curr = curr->next;
62     return curr;
63 }
64 #else // method 2 - rabbit and turtle
65 pNode half(plist p) {
66     pNode rabbit = begin(p);
67     pNode turtle = begin(p);
68     while (rabbit != last(n) && rabbit != end(n)) {
69         rabbit = rabbit->next;
70         turtle = turtle->next;
71     }
72     return turtle;
73 }
74 #endif

Enter N nodes to push back(-N for a value)? : 20000000

cpu: 0.502425 sec
FRONT  9651023 4758173 3704326 16651279  6777259 11969218  19705115  10643491  6832531 6872551
... 17043900 ...
15701436  12803651  14829314  1642048 9702131 10028511  14108818  6913411 16745381  11474343

Doubly Linked List(nodes:20000000, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop     0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      0(n)  n - n nodes per line
c - clear           0(n)
Command[q to quit]: q
Congratulations! It's half-time: cpu: 0.066205 sec

Cleared...

Joyful Coding==
Program ended with exit code: 0
```

Method 2 :

```
54 // For even numbers, it returns the first node of the second half.
55 // For example, for list [0, 1, 2, 3, 4, 5, 6, 7], it returns 4.
56 #if 0 // method 1 - slower version
57 pNode half(plist p) {
58     int N = size(p);
59     Node *curr = begin(p);
60     for (int i = 0; i < N / 2; i++)
61         curr = curr->next;
62     return curr;
63 }
64 #else // method 2 - rabbit and turtle
65 pNode half(plist p) {
66     pNode rabbit = begin(p);
67     pNode turtle = begin(p);
68     while (rabbit != last(n) && rabbit != end(n)) {
69         rabbit = rabbit->next;
70         turtle = turtle->next;
71     }
72     return turtle;
73 }
74 #endif

Enter N nodes to push back(-N for a value)? : 20000000

cpu: 0.508843 sec
FRONT  9481198 16704887  7766870 17922562  12213064  9832732 17910049  9693445 9058609 15691356
... 10272334 ...
427305 9638310 5226737 13620728  9212017 4102389 17247221  9783531 820325 12674501

Doubly Linked List(nodes:20000000, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop     0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      0(n)  n - n nodes per line
c - clear           0(n)
Command[q to quit]: q
Congratulations! It's half-time: cpu: 0.048796 sec

Cleared...
```

Step5. Swap_pairs

Odd number :

```
Command[q to quit]: y
FRONT  1  2  3  4  5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop     0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      0(n)  n - n nodes per line
c - clear           0(n)
Command[q to quit]: w
cpu: 2.9e-05 sec
FRONT  2  1  4  3  5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadoe8643ffbea667e6c

Even number :

```
cpu: 3.1e-05 sec
FRONT 1 2 3 4 5 6

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: w
cpu: 7e-05 sec
FRONT 2 1 4 3 6 5

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

Step6. Sorted())

Ascending :

```
cpu: 2.3e-05 sec
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: s
Sorted in ascending order
FRONT 1 2 3

Doubly Linked List(nodes:3, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)

Choose a position node: 7
FRONT 1 2 2 3 7

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: s
Sorted in ascending order
FRONT 1 2 2 3 7

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

Descending :

```
cpu: 5.2e-05 sec
FRONT 7 3 2 2 1

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: s
Sorted in descending order
FRONT 7 3 2 2 1

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afddfd7dbcaoaddoe8643ffbea667e6c

unsorted :

```
cpu: 3.6e-05 sec
FRONT  0  3  2  5  5  3

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: s
It is unsorted.
FRONT  0  3  2  5  5  3

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

Step7. Push_sorted()

Unsorted :

```
cpu: 5.2e-05 sec
FRONT  9  3  7  5  5  9  9  6

Doubly Linked List(nodes:8, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: z
The operation works in sorted list only.
FRONT  9  3  7  5  5  9  9  6

Doubly Linked List(nodes:8, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

Ascending :

```
cpu: 3.1e-05 sec
FRONT  3  5  5  7  9  9  9

Doubly Linked List(nodes:7, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse   0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: z
Enter a number to push: 8
cpu: 2.8e-05 sec
FRONT  3  5  5  7  8  9  9  9

Doubly Linked List(nodes:8, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
```

Descending :

```
cpu: 3.1e-05 sec
FRONT  9  9  9  7  5  5  3

Doubly Linked List(nodes:7, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: z
Enter a number to push: 6
cpu: 4.6e-05 sec
FRONT  9  9  9  7  6  5  5  3

Doubly Linked List(nodes:8, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

Step8. Unique()*

```
cpu: 3.6e-05 sec
FRONT  1  2  2  3  4  5  5  5  6

Doubly Linked List(nodes:9, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: u
cpu: 3.2e-05 sec
FRONT  1  2  3  4  5  6

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

Step9. Reverse()

```
cpu: 2.4e-05 sec
FRONT  1  2  3  4  5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
b - push back       0(1)  y - pop back  0(1)
B - push back N     0(n)  Y - pop back N 0(n)
i - push            0(n)  d - pop      0(n)
z - push sorted*    0(n)  e - pop all* 0(n)

s - sorted?         0(n)  r - reverse  0(n)
x - perfect shuffle* 0(n)  a - randomize 0(n)
u - unique*         0(n)  w - swap pairs 0(n)
t - show [ALL]      n - n nodes per line
c - clear           0(n)
Command[q to quit]: r
cpu: 3.2e-05 sec
FRONT  5  4  3  2  1

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front      0(1)  p - pop front  0(1)
```

// On my honor, I pledge that I have neither received nor provided improper assistance in the completion of this assignment.

// Signed: 조다빈 Student Number: 22000711

NOWIC

afdfdd7dbcaoadoe8643ffbea667e6c

Step10. Randomize()

```
cpu: 3.2e-05 sec
FRONT 1 2 3 4 5 6

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: a
cpu: 6.9e-05 sec
FRONT 3 6 4 5 2 1

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
```

Step11. Shuffle()*

Odd number :

```
cpu: 0.000104 sec
FRONT 1 2 3 4 5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: x
cpu: 3e-05 sec
FRONT 3 1 4 2 5

Doubly Linked List(nodes:5, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
```

Even number :

```
Command[q to quit]: b
Enter a number to push: 6
FRONT 1 2 3 4 5 6

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
i - push 0(n) d - pop 0(n)
z - push sorted* 0(n) e - pop all* 0(n)

s - sorted? 0(n) r - reverse 0(n)
x - perfect shuffle* 0(n) a - randomize 0(n)
u - unique* 0(n) w - swap pairs 0(n)
t - show [ALL] n - n nodes per line
c - clear 0(n)
Command[q to quit]: x
cpu: 2.9e-05 sec
FRONT 4 1 5 2 6 3

Doubly Linked List(nodes:6, show:HEAD/TAIL,10)
f - push front 0(1) p - pop front 0(1)
b - push back 0(1) y - pop back 0(1)
B - push back N 0(n) Y - pop back N 0(n)
```