Big Linked List

Create a project with 3 java files. The number file should just be the int node with the mynext and with a getval() method. The private data should be the int and the mynext. The second class is where all the linked list commands should be. The private data in this one should simply be a link to the first root in the list(myroot). The object should then be constructed in main. Main should then run through 200 numbers which are random numbers from 1 to 100. Your program should print each thing after it is called.

Your linked list class needs to -

| Add a node to the front of the list  Print the list  Add a node to the back of the list  Print the number of nodes in the list  Insert a node into a certain spot if  Check to see if your list is empty  Print the list  Get the first number  Get the last number  Get the number at a certain position  Get rid of the first node  Get rid of the last node  Count the number of nodes under 10  Get rid of a node at a certain point  Check to see if the list contains a  Create a reverse copy of the list  Create an iterator over the numbers  Average the set of numbers  Find the min of the numbers  Find the max of the numbers  Find the first index of the biggest  Find the last index of the biggest  Sort the numbers  Get rid of all 58s; return amount removed  Find the number of even numbers  Delete all odd numbers  Clear the entire list | public boolean addFront(int n)  public void printList()  public boolean addLast(int n)  public int getCount()  public boolean add(int n, int pos) (random num 1-200)  public boolean isEmpty()  public int getFirst()  public int getLast()  public int getByIndex(int pos) (-1 if illegal position)  public void removeFirst()  public void removeLast()  public int countLess10()  public void removeIndex(int pos)  public boolean contains(int n)  public IntLinkedList getReverse()  public Iterator<Integer> iterator()  public double avg()  public int min()  public int max()  public int indexOfMax()  public int lastIndexOfMax()  public void sort()  public int lose58()  public int getEvenCount()  public void removeOdds()  public void clear() |
| --- | --- |