# ICS 211 Practice Midterm 1

Clearly write your name on the **front** of this exam.

This exam is closed-book, closed-notes, closed-neighbor. No calculators are allowed. There are 50 points. Be sure to answer all parts of each question.

**Question 1** (10 points):

For the following code, answer each question.

public class Student extends Person implements HasGPA {

. . .

}

* What is the super class of **Student**? **Person** or **HasGPA**
* Must **Student** provide all the methods of **Person**?
* Must **Student** provide all the methods of **HasGPA**?
* Does **Student** inherit all the methods of **Person**?
* Does **Student** inherit all the methods of **HasGPA**?

Explain your answers below.

**Question 2** (5 points):

What is the Big-O for the following code in terms of n?

double sum = 0.0;

for (int m = 1; m < 0.5\*n; m++) {

sum += m;

for (int k = 0; k < 3 \* n; k += 2) {

if (sum % 2 == 0) {

sum -= k;

} else {

sum += 2 \* k;

}

}

}

**Question 3** (10 points):

Find the 3 syntax errors, 2 type errors, 1 initialization error and 1 precedence error in the following code.

public class Itter {

private E[] data;

private int index

public Itter() {

data = (E[]) new Object[](10);

index = 0;

}

public E next() {

if (!hasNext() {

throw NoSuchElementException();

}

int result = data[index];

index = index + 1 % capacity;

return result;

}

}

**Question 4** (5 points):

Write the DoublyLinkedList<E> **Node<E>** class.

**Question 5** (10 points):

Write the List interface:

**Question 6** (10 points):

Given the following code, implement the indexOf method.

public class LinkedList<E> {

private Node<E> head;

private int size;

. . .

/\*\*

\* Returns the index of the first occurrence of the

\* the specified element.

\* @param o the element.

\* @return the index of the element or -1.

\*/

public int indexOf(Object o) {