

Name: _____

CMSC132 Summer 2016 Midterm

True/False (10 points, 1 points each)

Indicate whether the statement is true or false.

- _____ 1. An interface can have public and private methods.
- _____ 2. The cost of insert in a sorted linked list is the same as remove which is $O(n)$.
- _____ 3. The “is-a” relationship between classes is called composition.
- _____ 4. This will correctly compute the length of a nonempty list:

```
int length = 0;
for (Node n = list.start; n != null; n = n.next) {
    ++length;
}
```
- _____ 5. Abstract classes can have constructors.
- _____ 6. Any comparison between non-comparable object types will raise a runtime exception
- _____ 7. Interfaces can have constructors.
- _____ 8. You can overload static methods in java.
- _____ 9. Unless there is a tail reference, appending a new node to the end of a linked list will always cost $O(n)$.
- _____ 10. In Java, the actual method executed is determined by the type of the object and not the type of the reference.

Multiple Choice(20 points, 2 points each)

Identify the choice that best completes the statement or answers the question.

- _____ 11. Choose correct equivalent statement of following code.

```
int[][] arr = new int[4][];
for (int i = 0; i < arr.length; i++)
    arr[i] = new int[3];
```
- | | | | |
|----|-------------------------------|----|-------------------------------|
| a. | int [][] arr = new int[4][]; | d. | int [][] arr = new int[4][3]; |
| b. | int [][4] arr = new int[4][]; | e. | None |
| c. | int [4][] arr = new int[4][]; | | |

Name: _____

_____ 12. Assume a singly linked list with "start" reference, and, in addition, a reference "current" that points to some Node in the list. How would you refer to the element after "current"?

- a. current.next
- b. list.next
- c. start.next
- d. next

_____ 13.

```
void print(Node n) {  
    while (n != null) {  
        System.out.print(n.data + ", ");  
        n = n.next;  
    }  
}
```



What will the output be of the code:
head = head.next.next;
print(head);

- a. 22, 33, 44,
- b. 33, 44, 55,
- c. 44, 55, 66,
- d. 22, 33, 44,
- e. 22, 33, 44, 55, 66 66666655

_____ 14. In the Sorted Bag, we used binary search in the contains method. Binary search algorithm can be applied to

- a. sorted linked list
- b. unsorted linear array
- c. sorted linear arrays
- d. unsorted linked list

_____ 15. The class "Parent" and its inherited class "Child" both implement a method "foo()", printing "parent" and "child" respectively. Which of the 4 choices below reflects the correct output of the following program:

```
Parent v1 = new Parent();  
Parent v2 = new Child();  
Child v3 = new Child();  
System.out.println(v1.foo()+" "+ v2.foo()+" "+ v3.foo());
```

- a. parent child child
- b. parent child parent
- c. parent parent parent
- d. child parent child

Name: _____

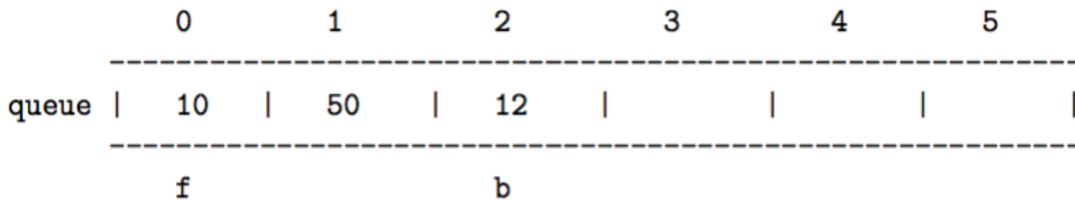
- _____ 20. A method defined in a superclass is redefined in a subclass with an identical method signature is called _____.
- a. Dynamic binding
 - b. Method overriding
 - c. Method overloading
 - d. Late binding

Short Answer (45 points, 3 points each)

21. The following code creates one array and one string object. How many references to those objects exist after the code executes? Is either object eligible for garbage collection?

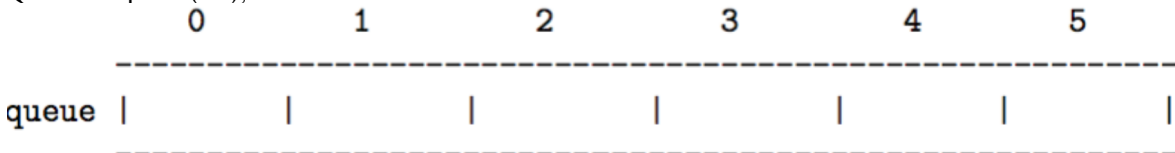
```
Person[] persons = new Person[10];
Person manager = new Person("alice");
persons[0] = manager;
manager = null;
Answer:
```

22. Given the following contents of a circular array implementation of a queue



Show the contents of the queue and locations of f (front) and b(back) after doing the following:

```
Queue.dequeue();
Queue.enqueue(7);
Queue.enqueue(8);
Queue.enqueue(20);
Queue.dequeue();
Queue.enqueue(11);
```



Name: _____

23. What is the output of the following program

```
public class Test{
    public String name;
    public static int score;
    public static void main(String[] args){
        Test t1 = new Test();
        Test t2 = new Test();
        t1.name = "C++";
        t1.score = 80;
        t2.name = "Ruby";
        t2.score = 85;
        System.out.println(t1.name);
        System.out.println(t2.name);
        System.out.println(t1.score);
        System.out.println(t2.score);
    }
}
```

Answer:

24. In a doubly linked list, P references a node, which is not the head neither the tail. Write code to delete the node. You are not allowed to start from Head.

Answer:

25. If list has more that one element, then what does the following code will do?

```
Node n = head;
while (n.next.next != null) {
    n = n.next;
}
n.next = null;
```

Answer:

Name: _____

```
26. Abstract class Parent{
    void foo(){
        System.out.println("parent");
    }
}
class Child extends Parent
{
    void foo(int x){
        System.out.println("child");
    }
}
```

Check the following if each statement is valid or invalid. Circle your answer

| | | |
|---------------------------|-------|---------|
| Parent p1 = new Parent(); | valid | invalid |
| Parent p2 = new Child(); | valid | invalid |
| Child c = new Child(); | valid | invalid |

| | | |
|------------|-------|---------|
| p2.foo(); | valid | invalid |
| p1.foo(5); | valid | invalid |
| c.foo(); | valid | invalid |
| c.foo(10); | valid | invalid |

27. Give the value of foo(5):

```
public static String foo(int n) {
    if (n <= 0) return "-";
    return foo(n-3) + n + foo(n-2) + n;
}
```

(hint: it returns a string)

28. We want to insert node t immediately after x. Is this code correct? If it is not correct, explain why?

```
x.next = t;
t.next = x.next;
```

Answer:

Name: _____

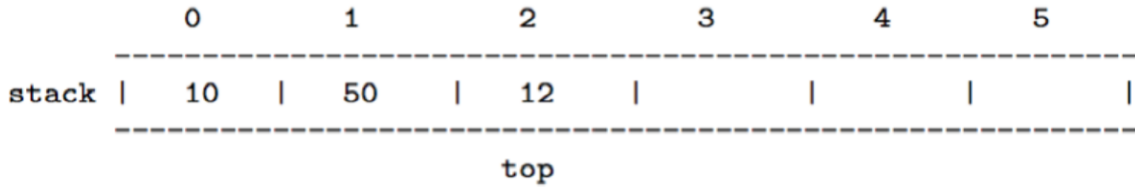
29. What does the following code fragment do to the queue q, which has string objects?

```
Stack<String> s = new Stack<String>();  
while(!q.isEmpty())  
    s.push(q.dequeue());  
while(!s.isEmpty())  
    q.enqueue(s.pop());
```

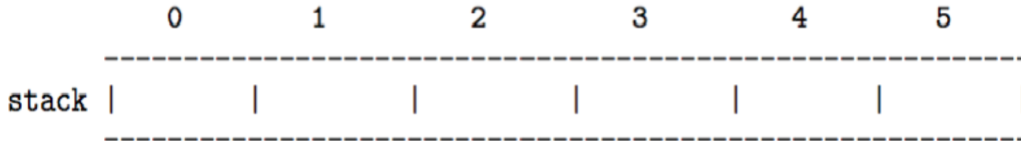
Answer:

.

30. Given the following contents of an array implementation of a stack



Show the contents of the stack and the location of top after doing the following
stack.pop();
stack.push(8);
stack.push(9);
stack.pop();
stack.push(11)



.

31. If a super-class and a subclass have the same field name, how would you access each of them in the sub-class?

Answer:

.

Name: _____

32. How does a java program destroy an object that it creates?

Answer:

.

33. Question: What's wrong with the following program?

```
Public class Person{
    public String name;
    public int age;
}

public class A {
    public static void main(String[] args) {
        Person person;
        person.name="alice";
        person.age = 22;
        System.out.println("Name: " + person.name);
    }
}
```

Answer:

.

34. Will the following code compile? IF YES, what would be the output? IF NO, what is the error?

```
class Data
{
    public Data(int i)
    {
        System.out.println(1);
    }
}
class File extends Data
{
    public File()
    {
        System.out.println(2);
    }
    public static void main(String[] args){
        File f = new File();
    }
}
```


Name: _____

35. We implemented the Bag with arrays and linked lists. List the pros and cons of those two kinds of Bags.

Answer:

Problem

36. (5 points)

```
public class Bag<E> implements Iterable<E>,Cloneable{
    protected E[] items;
    protected int N;//number of items in the bag
    protected int capacity = 10;
    public Bag(){
        items = (E[])new Object[capacity];
    }
}
```

The resizable Bag class uses an array to store its elements. The variable “items” references the array. Write a public method “min” for Bag class. This method returns the minimum item in the bag if the Bag is not empty. It returns “null” if the bags empty.

Answer:

Name: _____

37. (5 points) Write a method "Node half(Node head)" that returns the second half of a given linked list. If the list length is odd, middle node belongs to the FIRST half. "head" references the first node of the list. The length of the list is not given. for example: if the list is head->1->7->13->2->4, you return head->13->2->4.
Node half(Node head)

Name: _____

38. (10 points) The singly-linked List class below stores integers. The list is not sorted. Head is null when a list is empty, which means that the list does not have a dummy head node. Moreover, the list is not circular. The List and Node classes only have the fields shown.

```
public class List {
    private static class Node {
        int data;
        node next;
    }
    private Node head = null;
    public void removeNode(int index){TO DO}
}
```

Write the method "public void removeNode(int index)". It should modify its current object list by removing the node for given the index. If the length of the list is smaller than the given index, then the list should not be modified.

For example, if a list named exampleList contains the elements 1, 3, 7, 6, 8 and 5, in this order, then after calling exampleList.removeNode(4) it would contain just 1, 3, 7, 8 and 5. (the 6 was removed).

Your method should work properly, without errors, regardless of the length of the list, which may be empty or have any number of elements. To get full credit, your method:

Should be efficient, meaning it should not do unnecessary work. May not call any other methods of the List or Node classes. May not copy data values from one node to another, or instantiate any nodes.

Comments are optional, but if you have time to write short ones, it might help you get more partial credit in case of mistakes.

Name: _____

39. (5 points) Write the countOccurrences method with prototype

“public int countGreater(E target)” that returns the number of items in the bag that are greater than the given element. Assume that null data elements are not allowed in the bag.

Answer: