COMP 2404

Midterm Exam

[out of 50 marks]

Duration: 80 minutes

Authorized Memoranda: NONE

Name: [redacted]

Multiple Choice Questions (circle ONE answer for each question) [10 marks]

1. Given the program in Figure 1, which of lines (24) to (26) are valid and will not result in an error:

   a. (24)
   b. (25)
   c. (26)
   d. (24) and (25)

2. Given the program in Figure 1, which of lines (28) to (30) are valid and will not result in an error:

   a. (28) and (29)
   b. (28)
   c. (30)
   d. (29)

3. Given the program in Figure 1, which of lines (32) to (34) are valid and will not result in an error:

   a. (32)
   b. (33)
   c. (34)
   d. (32) and (34)

4. Given the program in Figure 1, which of lines (36) to (38) are valid and will not result in an error:

   a. (36) and (37)
   b. (37)
   c. (38)
   d. none

5. Given the program in Figure 1, which of lines (40) to (42) are valid and will not result in an error:

   a. (40) and (41)
   b. (41)
   c. (42)
   d. none

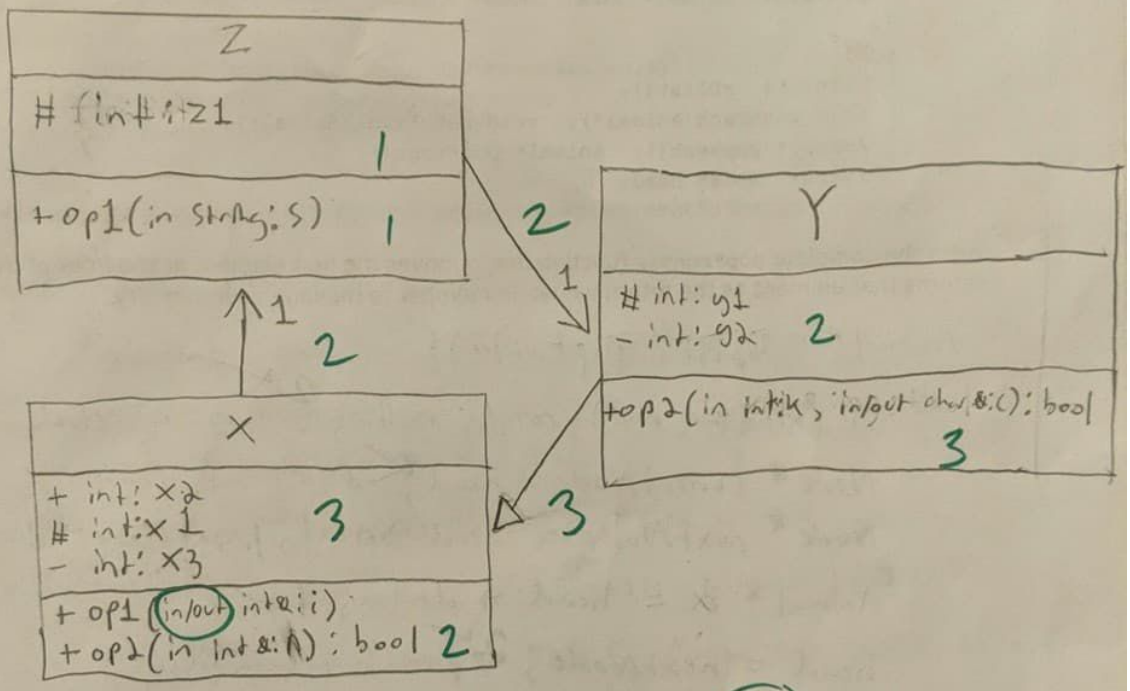
```
01  class Z {
02    public:     void op1(string s) { s = "xyz"; }
03    protected:  int z1;
04    private:    Y   y1;  };
05
06  class X {
07    public:     void op1(int& i) { i = 99; }
08                bool op2(int& n) { if (n > 0) return true; else return false; }
09                int  x2;
10                Z    z1;
11    protected:  int  x1;
12    private:    int  x3;  };
13
14  class Y : public X {
15    public:
16      bool op2(int k, char& c) { if (k < 10 && c > 0) c = 'J'; return true; }
17    protected:  int y1;
18    private:    int y2;  };
19
20  int main() {
21    X x;  Y y;  Z z;
22    string tmp = "hello";  int n1 = 12;  int n2 = 34;  char c  = 'C';
23
24    y.op2(n1);
25    y.op2(n1, c);
26    x.op2(n1, c);
27
28    y.x1 = 0;
29    y.x2 = 0;
30    x.y1 = 0;
31
32    x.z1.op1(tmp);
33    z.x1.op2(n1);
34    z.y1.op2(n1,c);
35
36    z.y1.op2(n2);
37    z.y1.op2(n2, c);
38    y.z1.op1(tmp);
39
40    y.z1.z1 = 0;
41    y.z1.op1(tmp);
42    z.y1.y1 = 0;
43
44    return 0;
45  }
```

no x1 in Z

Figure 1

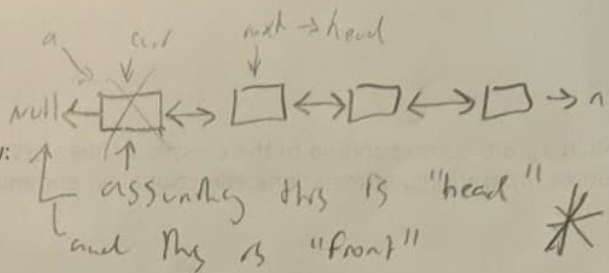
ML Question **[20 marks]**

6. Draw the UML diagram corresponding to the classes defined in Figure 1. You must show all classes, and all attributes, operations, associations, directionality, and multiplicity, where applicable:



19

19



Programming Questions [20 mar

7. Given the class definition below:

```
class Dlist  {
  class Node {
    friend Dlist;
    private:  Animal* data;  Node*   prev;  Node*   next;
  };
  public:
    Dlist();  ~Dlist();
    void pushBack(Animal*);  void pushFront(Animal*);
    Animal* popBack();  Animal* popFront();
  private:  Node* head;
};
```

write the complete `popFront()` function that removes the first element at the front of the list and returns that element as the return value. Remember to manage your memory.

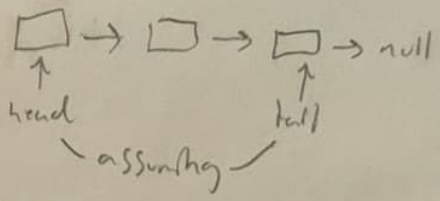
[10 marks]

```
Animal* DList::popFront(){
    if (head == null){return null;} 2 // base err check
    Node* currentNode = head; // start at front
    Node* nextNode = head->next; | nextNode->prev = null; * |
    Animal* a = head->data; | ↑ // disconnect head
    head = nextNode; 2 // point to new head
    delete currentNode; | // current still holds original head
    return a; |
}
```

10

10

front



Given the class definition below:

```
class Tlist  {
  class Node {
    friend Tlist;
    private:  Animal* data;  Node* next;
  };
  public:
    Tlist();  ~Tlist();
    void pushBack(Animal*);  void pushFront(Animal*);
    Animal* popBack();  Animal* popFront();
  private:  Node* head;  Node* tail;
};
```

write the complete `pushBack()` function that adds the given element to the back of the list.

[10 marks]

```
void Tlist::pushBack(Animal* a){
  //Node* currNode = tail;  <- don't need
  Node* addNode;
  addNode->data = a;                  } declare new node
  tail->next = addNode;  // add to end
  tail = addNode;  // tail points to new end
  addNode->next = null;  // new end points to null
}
```

2 2 1

4

4