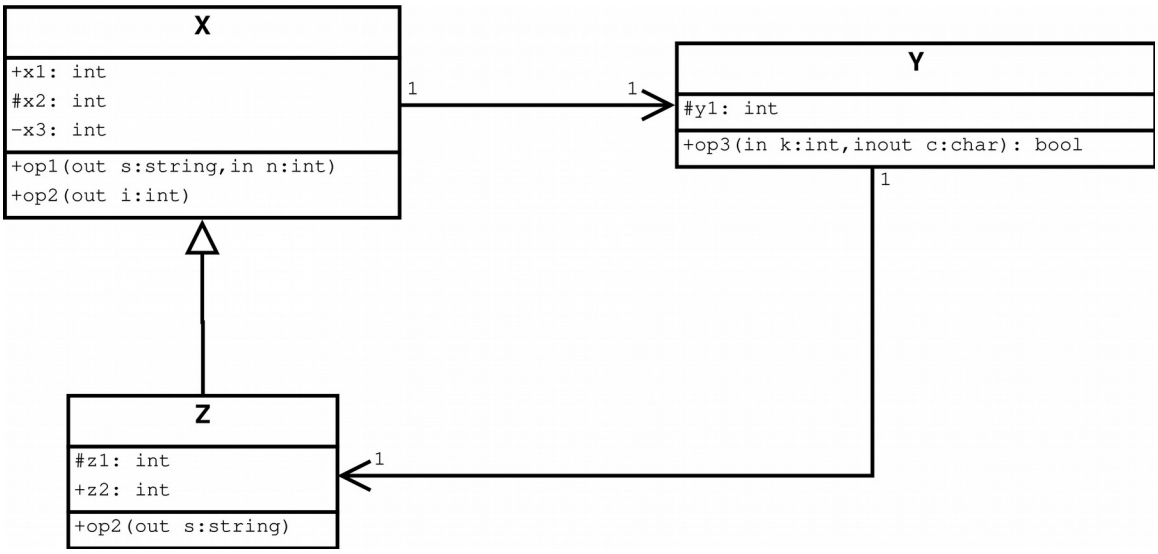
COMP 2404

Midterm Exam Solution -- Version 1

1. [2 marks] c
2. [2 marks] b
3. [2 marks] a
4. [2 marks] b
5. [2 marks] d

6. [20 marks]



Grading:

- 3 marks: X attributes with correct access specifier, 1 mark each
- 3 marks: X operations, 1 mark each correct parameter
- 2 marks: Z attributes with correct access specifier, 1 mark each
- 1 mark: Z operation
- 1 mark: Y attribute with correct access specifier
- 3 marks: Y operations, 1 mark each correct parameter and return type
- 3 marks: inheritance relationship between X and Z
- 2 marks: composition relationship between X and Y, with directionality and multiplicity
- 2 marks: composition relationship between Y and Z, with directionality and multiplicity
- -1 mark: each object shown as attribute

7. [8 marks]

```
void Dlist::pushFront(Animal* critter)
{
  Node* newNode;

// 4 marks for allocating and initializing new node
// -- 2 marks allocating node
// -- 1 mark initializing data
// -- 1 mark initializing prev
  newNode = new Node;
  newNode->data = critter;
  newNode->prev = 0;
  newNode->next = 0;

// 1 mark for linking the rest of the list
  newNode->next = head;

// 2 marks for setting old head's prev, if not null
  if (head != 0)
    head->prev = newNode;

// 1 mark for setting head to new node
// -1 mark for deleting new node
  head = newNode;
}
```

8. [12 marks]

```
Animal* Tlist::popBack()
{
  Animal* goner;
  Node*   currNode;
  Node*   prevNode;

// 2 marks for dealing with empty list case
  if (head == 0)
    return 0;

// 2 marks for finding the end of the list
  currNode = head;
  prevNode = 0;
  while (currNode != tail) {
    prevNode = currNode;
    currNode = currNode->next;
  }
// 1 mark for saving last element
  goner = tail->data;
// 1 mark for deleting tail node
  delete tail;

// 3 marks for dealing with single element case
  if (prevNode == 0) {
    head = tail = 0;
  }
// 2 marks for dealing with regular case
  else {
    prevNode->next = 0;
    tail = prevNode;
  }
// 1 mark for returning last element
  return goner;  }
```