# COMP 2404
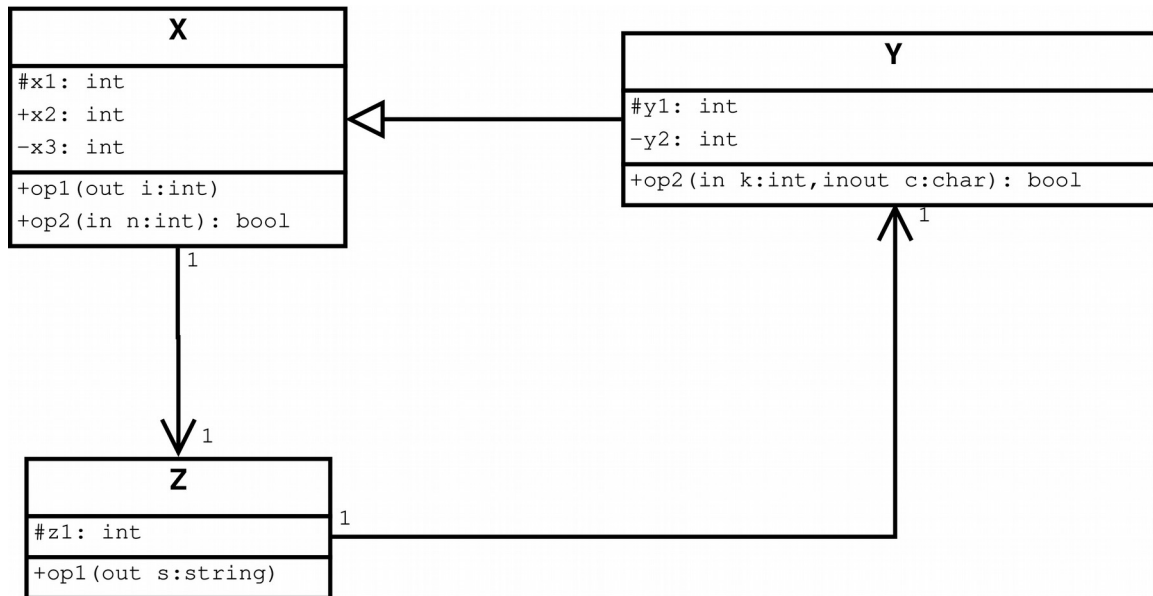## Midterm Exam Solution -- Version 3

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

6.  [20 marks]

```
        ┌─────────────────────┐              ┌──────────────────────────────────────┐
        │          X          │              │                  Y                   │
        ├─────────────────────┤              ├──────────────────────────────────────┤
        │ #x1: int            │ ◁────────────│ #y1: int                             │
        │ +x2: int            │              │ -y2: int                             │
        │ -x3: int            │              ├──────────────────────────────────────┤
        ├─────────────────────┤              │ +op2(in k:int,inout c:char): bool    │
        │ +op1(out i:int)     │              └──────────────────────────────────────┘
        │ +op2(in n:int): bool│                                      △ 1
        └─────────────────────┘                                      │
                  │ 1                                                 │
                  │                                                   │
                  ▼ 1                                                 │
        ┌─────────────────────┐                                      │
        │          Z          │                                      │
        ├─────────────────────┤ 1                                    │
        │ #z1: int            │──────────────────────────────────────┘
        ├─────────────────────┤
        │ +op1(out s:string)  │
        └─────────────────────┘
```

Grading:

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

7.  [10 marks]

```cpp
Animal* Dlist::popFront()
{
  Animal* goner;
  Node*   newHead;

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

// 1 mark for saving the last element
  goner   = head->data;

// 1 mark for saving next head
  newHead = head->next;

// 1 mark for deleting first node
  delete head;

// 2 marks for setting new head
  head    = newHead;

// 2 marks for setting new head prev to null
  newHead->prev = 0;

// 1 mark for returning the last element
  return goner;
}
```

8.   [10 marks]

```
void Tlist::pushBack(Animal* critter)
{
  Node* newNode;

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

// 3 marks for dealing with empty list case
  if (head == 0) {
    head = tail = newNode;
  }
// 3 marks for dealing with regular case
  else {
    tail->next = newNode;
    tail = tail->next;
  }
}
```