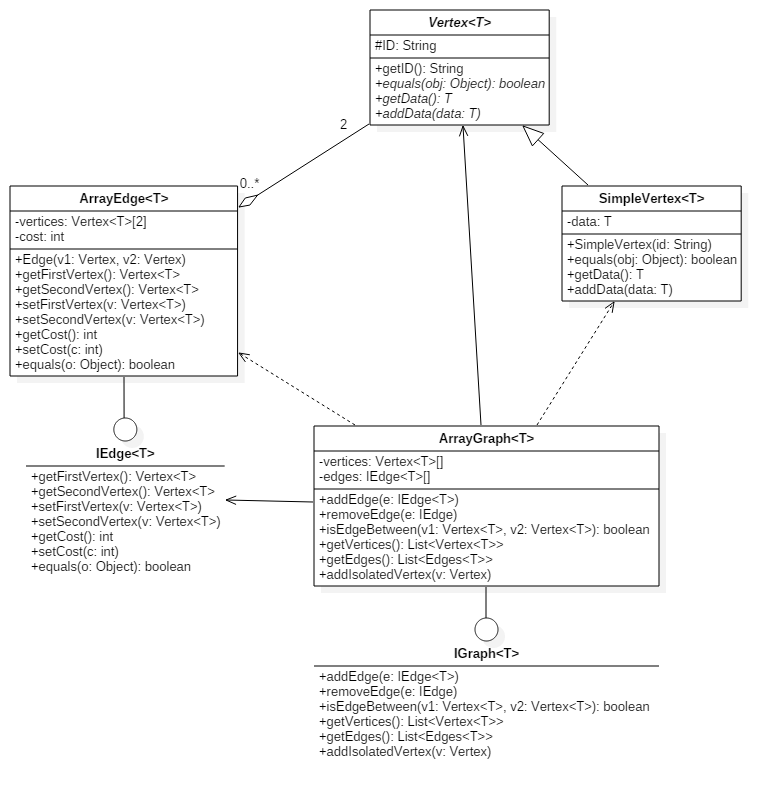
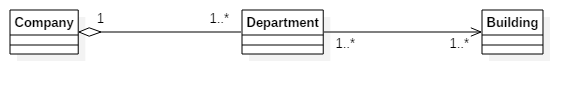
**P1.** Prepare a class model to describe undirected graphs. An undirected graph consists of a set of vertices and a set of edges. Edges connect pairs of vertices. Your model should capture only the structure of graphs (i.e. connectivity).

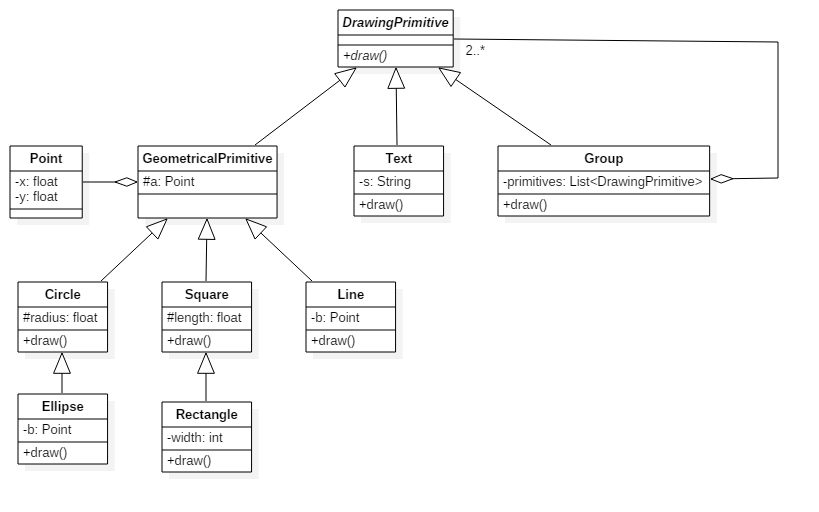


Clasa abstracta Vertex<T> reprezinta un nod un grafului, care este identificata printr-un ID. Un nod al grafului poate stoca o data de tipul generic T. Clasa SimpleVertex<T> mosteneste si implementeaza metodele abstracte ale clasei Vertex<T>. Interfata IEdge<T> descrie comportamentul unei muchii. Clasa ArrayEdge<T> implementeaza interfata IEdge<T> continand un vector de 2 elemente de tipul Vertex<T>. Muchiilor I se poate asocial si un cost daca este cazul (implicit este 0). Interfata IGraph<T> contine metodele principale ale unui graf, si anume adaugarea/stergerea unei muchii, adaugarea unui varf izolat, verificarea existentei unei muchii si intoarcerea muchiilor si a varfurilor care alcatuiesc graful. Clasa ArrayGraph<T> implementeaza interfata IGraph<T> cu ajutorul unor tablouri.

**P2.** Consider the following specification: *“A company consists of several departments. Each department is located in one or more buildings. ”*. Draw a class diagram to model the concepts above (no attributes and operations, just classes and the relationships between them including multiplicities).



**P3.** Prepare a class diagram for a graphical document editor that supports grouping. Assume that a document consists of several sheets. Each sheet contains drawing objects, including text, geometrical objects, and groups. A group is simply a set of drawing objects, possibly including other groups. A group must contain at least two drawing objects. A drawing object can be a direct member of at most one group. Geometrical objects include circles, ellipses, rectangles, lines and squares.

Clasa abstracta DrawingPrimitive contine o singura metoda: draw. Aceasta metoda este folosita pentru afisarea primitive grafice pe ecran. Clasele GeometricalPrimitive, Text si Group mostenesc clasa DrawingPrimitive.

Clasa GeometricalPrimitive este abstracta, unind singurul atribut al primitivelor grafice geometrice: punctul sub forma unei variabile instant de tipul Point. Clasa Circle mosteneste clasa abstracta GeometricalPrimitive si implementeaza metoda draw, este extinsa de clasa Ellipse care la randul ei adauga inca un punct ca atribut. Clasa Square adauga la GeometricalPrimitive lungimea si implementeaza metoda draw, iar clasa Rectangle este si mai specifica adaugand si latimea. Clasa Line adauga la GeometricalPrimitive inca un punct si implementeaza metoda draw.

Clasa Text contine un singur atribut (un sir de caractere). In cazul clasei Group am folosit sablonul de proiectare Composite: Group are o lista de obiecte DrawingPrimitive (superclasa ei).

**P6.** Draw an UML diagram for the following C++ segment of code:

class Person

{

public:

Person(const char\* aName);

void speak();

void drive();

private:

char\* name; // Persons Name

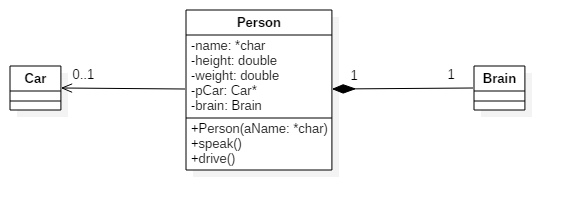
Brain brain; // persons brain

Car\* pCar; // Car owned by the person

double height;

double weight;

};



**P7.** Given the following code:

class Memory {…} // Assume a copy constructor is provided for this class.

class Memory1 extends Memory {…} // Assume a copy constructor is provided.

class Computer

{

private Memory theMemory;

public Computer(Computer another)

{

if (another.theMemory instanceof Memory1)

theMemory = new Memory1(another.theMemory);

else

theMemory = new Memory(another.theMemory);

}

}

Draw a UML class diagram showing the relationship between these classes.

