Exercise A:

Type D

Node

~keyM : Integer
~itemM : D
~nextM : Node<D>

+Node()
+Node(itemA: D, keyA: Integer, nextA: Node<D>)

Cloneable

Point

-x: double
-y: double

+Point(a: double, b: double)
+getx() : const double()
+gety() : const double()

Shape

#origin: Point
#shapeName: string

+Shape(x_origin: double, y_origin: double, name: string)
~Shape()
+Shape(source: const Shape&)
+getOrigin(): Point
+getName(): string
+distance(the_shape: Shape&, other: Shape&): double
+*area(): double*

Exercise B:

Vector   **Department**   **Professor**

1-*   1   1   1-*

+Head

1

1-*   1

Student   **Section**   **Course**

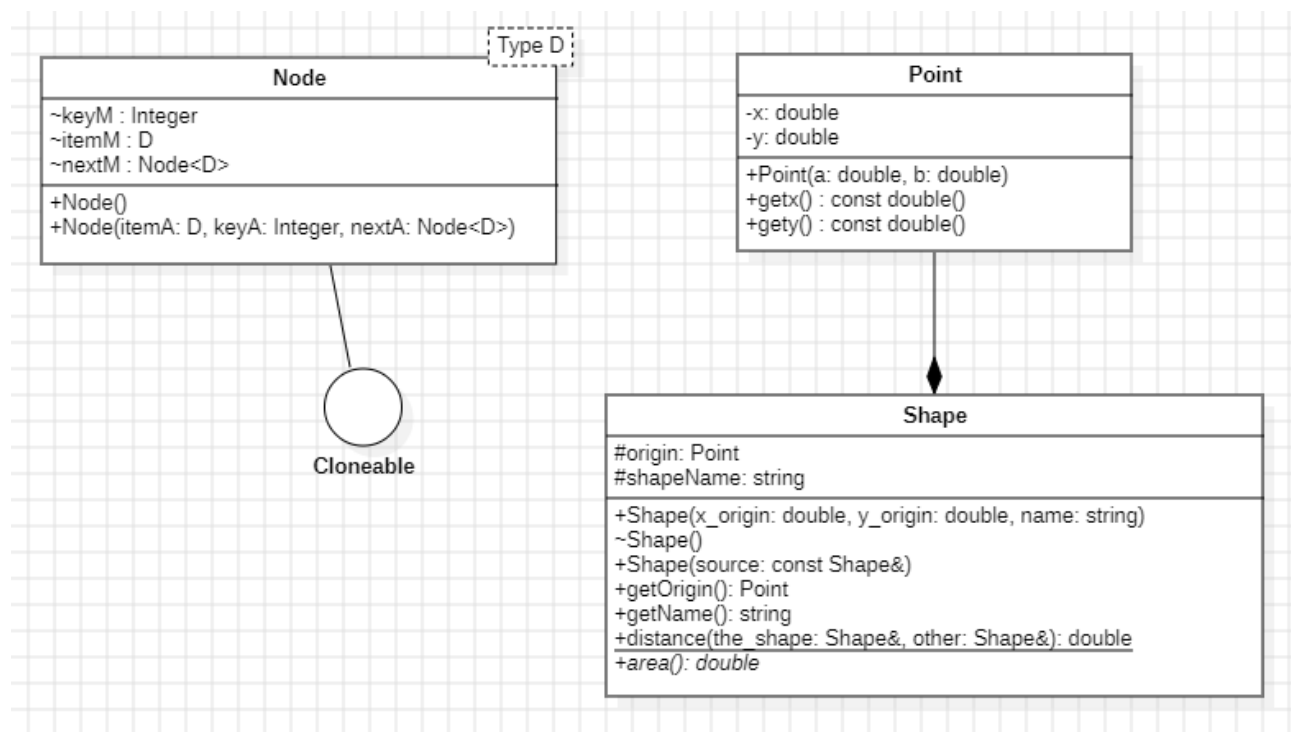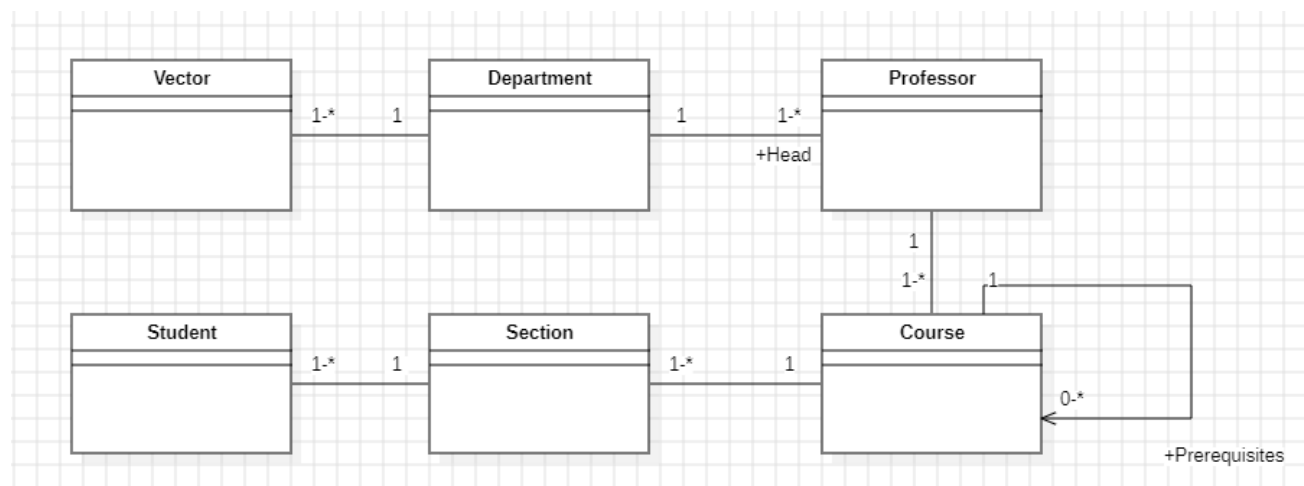1-*   1   1-*   1

0-*

+Prerequisites

Exercise C:

```cpp
#include <string>
#ifndef EXC_H
#define EXC_H
using namespace std;

class Moveable{
    public:
        virtual void forward();
        virtual void backward();
};

class Resizeable{
    public:
        virtual void enlarge(int n);
        virtual void shrink(int n);
};

class Vehicle : public Moveable, public Resizeable{
    protected:
        string name;
    public:
        Vehicle(string name);
        void move();
        void forward() override;
        void backward() override;
        void enlarge(int n) override;
        void shrink(int n) override;
};

class Car : public Vehicle {
    private:
        int seats;
    public:
        void turn();
};


#endif
```

Exercise D:

Point
- -x_coordinate: double
- -y_coordinate: double
- -counter: int
- -id: String
- +Point(a: double, b: double)
- +toString(): String
- +get(): double
- +set(): double
- +setX(newvalue: double)
- +setY(newvalue: double)
- +distance(other: Point): double
- +distance(that: Point, other: Point): double
- +count(): int
- +clone(): Object
- +main(args: String)

Sun
- -min_distance: double
- -planets: Planet[]
- -num_planets: int
- -counter: int
- +Sun(c: Point, radius: double, name: String, distance double, planets: Planet[])
- +toString(): String
- +setName(newname: String)
- +area(): double
- +perimeter(): double

Planet
- -counter: int
- -moons: Moon[]
- -num_moons: int
- -orbital_distance: double
- +Planet(c: Point, radius: double, name: String, distance double, moons: Moon[])
- +toString(): String
- +area(): double
- +clone(): double
- +setName(newname: String)
- +perimeter(): double

MoreThanLimitException
- +MoreThanLimitException(msg: String)

CelestialBody
- #center: Point
- #radius: double
- #name: String
- -counter: int
- #id: String
- +CelestialBody(p: Point, radius: double, name: String)
- +clone(): Object
- +toString(): String
- +getRadius(): double
- +setRadius(radius: double)
- +getName(): String
- +setName(newname: String)
- +getOid(): String

Moon
- -counter: int
- -distance_from_planet: double
- +Moon(p: Point, radius: double, name: String, pl_distance double)
- +toString(): String
- +area(): double
- +setRadius(radius: double)
- +setName(newname: String)
- +perimeter(): double

Shape
- #name: String
- #id: String
- -counter: int
- -origin: Point
- +Shape(x_origin: double, y_origin: double, name: String)
- +clone(): Object
- +setName(name: String)
- +getOid(): String
- +getOrigin(): Point
- +getName(): String
- +distance(other: Shape): double
- +distance(the_shape: Shape, other: Shape): double
- +move(dx: double, dy: double)
- +toString(): String
- +count(): int

Circle
- -radius: double
- -id: counter: int
- +Circle(x_origin: double, y_origin: double, radius: double, name: String)
- +clone(): Object
- +getOid(): String
- +getRadius(): double
- +setName(newname: String)
- +area(): double
- +perimeter(): double
- +volume(): double
- +toString(): String

«interface»
Accessible
- +getName(): String
- +setName(name: String)
- +getOid(): String

Cloneable

«interface»
Calculable
- +area(): double
- +perimeter(): double

LabImplementable
- +CELESTIAL = 100 (readOnly)
- +SHAPE = 200 (readOnly)
- +CIRCLE = 2000 (readOnly)
- +PLANET = 10000 (readOnly)
- +SUN = 1000 (readOnly)
- +MOON = 100000 (readOnly)
- +getRadius(): double
- +setRadius(value: double)

MoonRadiusException
- +public MoonRadiusException(a: Accessible, msg: String)

CloneNotSupportedException

RadiusException
- +public RadiusException(a: Accessible, msg: String)

[relations: «throws», «uses», «extends», «implements», «interface»]

https://drive.google.com/file/d/10faGcDWYbDZtEi0jtGHJlgvHWkd741xx/view?usp=sharing **If you need a better view**