Tuple (Point/Vector)

```
  x, y, z, w : Double
----------------------------
  dot(Vector) -> Double
  cross(Vector) -> Vector
  reflectOver(Vector) -> Vector
  +
  -
  * scalar
  / scalar
```

Intersection

```
  object : Object
  t : Double
  ray : Ray
--------------------------
  generateHitRecord() -> Intersection
```

HitRecord (struct)

```
  hitPoint : Point
  normal : Vector (unit)
  eye : Vector (unit)
  overPoint : Point
  underPoint : Point
  isInside : bool
```

Matrix

```
  elements : Array, List, Vector/1 or 2D
-----------------------------
  calculateDeterminant() -> Double
  cofactor(i, j) -> Double
  minor(i, j) -> Double
  submatrix(i, j) -> Matrix
  inverseMatrix() -> Matrix
  transposeMatrix() -> Matrix
```

Light

```
  position : Point
  color : Color
--------------------------------
  virtual directionFromPoint(Point) -> Vector
  virtual distanceFromPoint(Point) -> Double
```

PointLight : Light

```
  directionFromPoint(Point) -> Vector
  distanceFromPoint(Point) -> Double
```

DirectionalLight : Light

```
  direction : Vector
------------------------------------------
  virtual directionFromPoint(Point) -> Vector
  virtual distanceFromPoint(Point) -> Double
```

Material

```
  color : Color
  diffuse : Double
  ambient : Double
  specular : Double
  shininess : Double
  reflectivity : Double
  transparency : Double
  indexOfRefraction : Double
------------------------------------------
  colorAtPoint(HitRecord, World) -> Color
```

Shape

```
  material : Material
  transform : Matrix
-----------------------------------
  translate(x, y, z)
  scale(x, y, z)
  rotateAroundX(Double)
  rotateAroundY(Double)
  rotateAroundZ(Double)
  virtual intersectionsWith(Ray) -> Intersections[]
  virtual normalAt(Point) -> Vector(unit)
```

Sphere : Shape

```
  intersectionsWith(Ray) -> Intersections[]
  normalAt(Point) -> Vector(unit)
```

Plane : Shape

```
  intersectionsWith(Ray) -> Intersections[]
  normalAt(Point) -> Vector(unit)
```

Triangle : Shape

```
  vertices : Point[]
---------------------------------------
  intersectionsWith(Ray) -> Intersections[]
  normalAt(Point) -> Vector(unit)
```

MeshTriangle : Triangle

```
  indices : Int[]
  mesh : Mesh
```

Mesh : Shape

```
 meshTriangles : MeshTriangle[]
 vertices : Point[]
```

Color

```
 r, g, b : Double
-----------------------------------------
 intRed() -> Int
 intGreen() -> Int
 intBlue() -> Int
 +
 *
```

Ray

```
 origin : Point
 direction : Vector
------------------------------------------
 pointAtT(Double) -> Point
 transform(Matrix) -> Ray
```

Scene

```
 objects : Shape[]
 lights : Light[]
-------------------------------
 addObject(Shape)
 addLight(Light)
 intersectionsWith(Ray) -> Intersection[]
 colorAtIntersection(Intersection, Intersection[]) -> Color
 colorForRay(Ray) -> Color
```

Canvas

```
  width : Int
  height : Int
  pixels : Color[]
--------------------------
  toPPM() -> string
  writePPM(filename)
```