Question 1

1. (*5 points*) What is the class hierarchy of the interpolators? Please write your answer in the following format:

BaseClass
    ChildClass1
        GrandchildClass1
    ChildClass2
        GrandchildClass2
        GrandchildClass3

*The base class is **AInterpolatorVec3***

Answer
According to the format above

AInterpolatorVec3 //base class
    ALinearInterpolatorVec3 //ChildClass1
    ACubicInterpolatorVec3 //ChildClass2
        ABernsteinInterpolatorVec3 //GrandchildClass1
        ACasteljauInterpolatorVec3 //GrandchildClass2
        AMatrixInterpolatorVec3 //GrandchildClass3
        AHermiteInterpolatorVec3 //GrandchildClass4
        ABSplineInterpolatorVec3 //GrandchildClass5
        AEulerCubicInterpolatorVec3 //GrandchildClass6
    AEulerLinearInterpolatorVec3 //ChildClass3

Question 2

2. *What does the function vec3 **ABernsteinInterpolatorVec3**::**interpolateSegment**() do?*

The function initially inherits from the virtual function of
AInterpolatorVec3::interpolateSegment (
        const std::vector<ASplineVec3::Key>& keys,
        const std::vector<vec3>& ctrlPoints,
        int segment, double u)
which using given keys, control points, current segment start index, and the time, compute an interpolated value. Then it passes to the child class ACubicInterpolatorVec3 and to its grandchild class ABernsteinInterpolatorVec3.

In the specific case of a child class defined function, ABernsteinInterpolatorVec3:: interpolateSegment(), it uses the input elements defined above to

1. Get 4 control points from the ctrlPoints vector
2. Then compute the interpolated value f(u) point using Bernstein polynomials, and return the result as a vec3 datatype