

---

**算法 1** vertexfragment
 

---

输入数据:

vertexInf={ *position*, *UV*, *skinIndex* }//当前顶点的信息

instanceObjInf={ //实例化对象的信息

*matrix*,

*animationStyle*,//动画的类型和速度

*textureStyle*,//贴图的类型和色调

}

uniformInf={ //所有实例化对象共用的数据

*skelontonDate*,//骨骼数据

*time*,//时间用于计算当前的帧序号

}

**function** main()

*judgeArea*(*vertexInf.position*); //判断当前顶点处于身体的哪个部位

*matrix1* = *computeAnimationMatrix*(*uniformInf.time*,*instanceObjInf.animationStyle.speed*,  
*uniformInf.skelontonDate*,*instanceObjInf.animationStyle.type*,*vertexInf.skinIndex*);

*matrix2* = *computeInstanceObjMatrix*();

*glPosition* = *projectionMatrix* \* *modelViewMatrix* \* *matrix2* \* *matrix1* \* *vec4*(*position*, 1.0);

**end function**

**function** *computeAnimationMatrix*(*time*,*speed*,*skelontonDate*,*type*,*skinIndex*)

**if** *bone*(*vertexInf.skinIndex*)*haveanimation* **then** //对应的骨骼有动画

*frameIndex* = *time* \* *speed* *mod* (*frameIndexMax* + 1);

*address0* = *addressGet1*(*skinIndex*,*type*, *frameIndex*);

**else**//对应的骨骼没有动画

*address0* = *addressGet2*(*skinIndex*,*type*);

**end if***result* = *getMatrix*(*skelontonDate*, *address0*);

**return** *result*

**end function**

输出数据:

*glPosition* //当前顶点对应屏幕上的位置

*sendFragmentShader*={ //传递给 *FragmentShader* 的信息

*areaType*,//当前点所在的区域

*UV*, *textureType*,//贴图类型

*color*,//由于色调调整

}

---