

Winter Semester 2017/18

Assignment on Virtual Reality and Physically-Based Simulation - Sheet 3

Due Date November 26. 2017

Exercise 1 (Laggy jump'n run, 3 Credits)

In this exercise we will test the influence of latency, to our performance in computer games. From the CGVR web site (http://cgvr.cs.uni-bremen.de/teaching/vr_1718/index.shtml), you can download a simple jump and run game with adjustable input delay. The goal is to complete the course from start to finish, as fast as possible and without falling down.

- Each member of your group should complete the game with at least three different delays.
- Write down the number of starts needed for the first successful run, and the time for the fastest completion. Do this for each delay (note your delay) and for each member of your group. Sent the results to me in the following format:

```
delay , starts , time  
10 , 4 , 60  
20 , 2 , 42  
30 , 10 , 40  
10 , 1 , 550  
20 , 1 , 66  
30 , 3 , 67
```

Exercise 2 (Unreal C++, 5 Credits)

This time the goal is to create an Actor completely in C++. Add a new C++ Actor to the laggy jump project that moves a cube up and down during the game. Make the movement speed or range adjustable in the editor through a property. You can see a possible result in this video.¹

From the Unreal documentation this part seems especially useful to get started with C++.² You can add the cube mesh to your actor like this:

```
static ConstructorHelpers::FObjectFinder<UStaticMesh>  
    mesh(TEXT("StaticMesh'/Game/Geometry/Meshes/1M_Cube_Chamfer'"));  
StaticMesh = CreateDefaultSubobject<UStaticMeshComponent>(TEXT("StaticMesh1"));  
StaticMesh->SetStaticMesh(mesh.Object);
```

¹ http://cgvr.informatik.uni-bremen.de/teaching/vr_1718/uebungen/03_result.mp4

² <https://docs.unrealengine.com/latest/INT/Programming/Introduction/index.html>