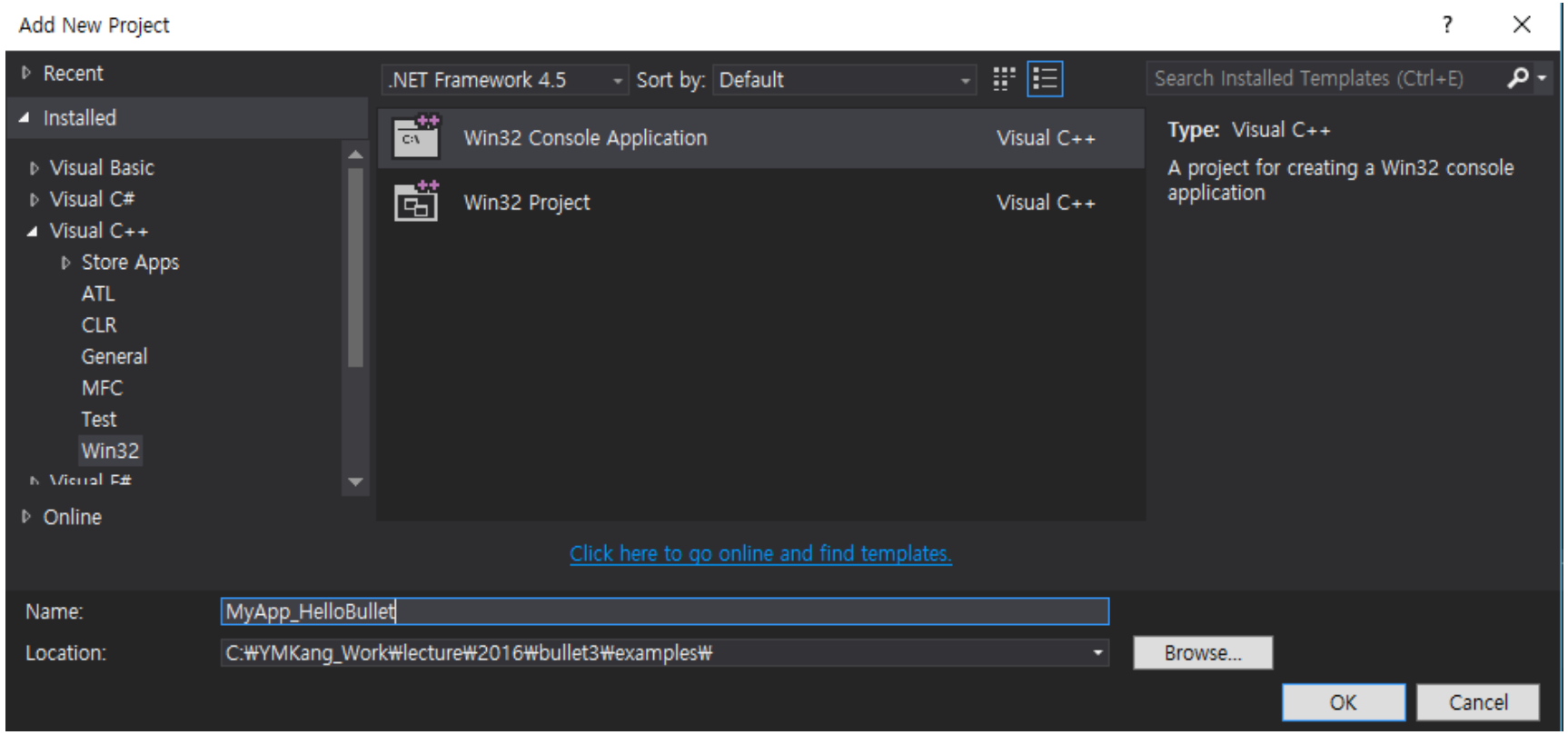
Bullet3 Tutorial

강영민

Hello Bullet

- Create an application
  - Win32 Console Application

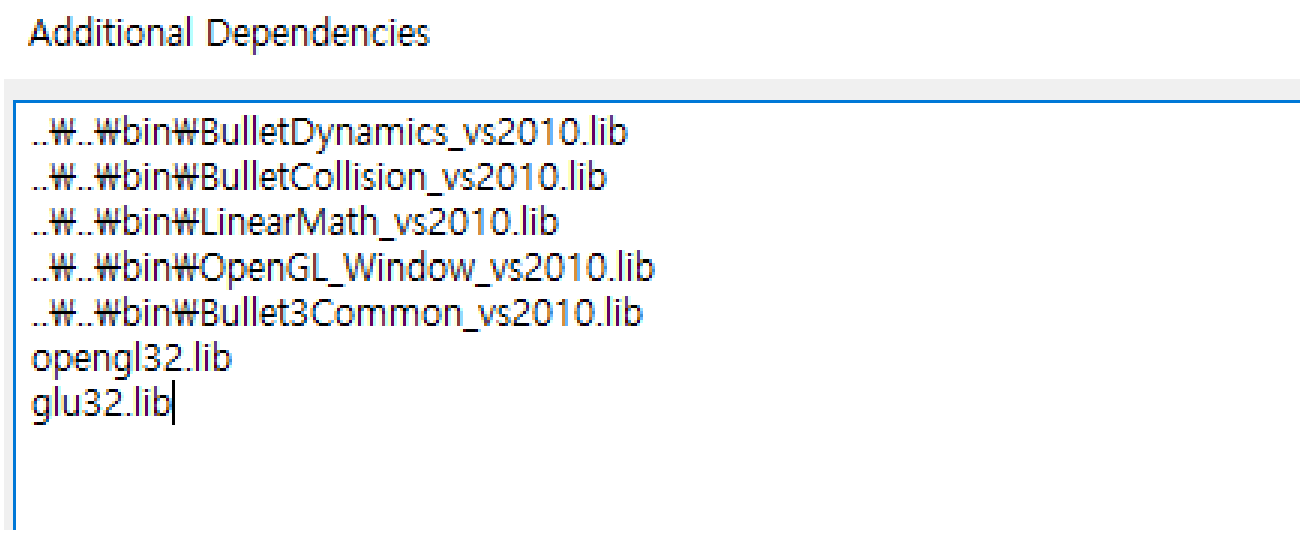


Add a source code

- MyApp_HelloBulletMain.cpp

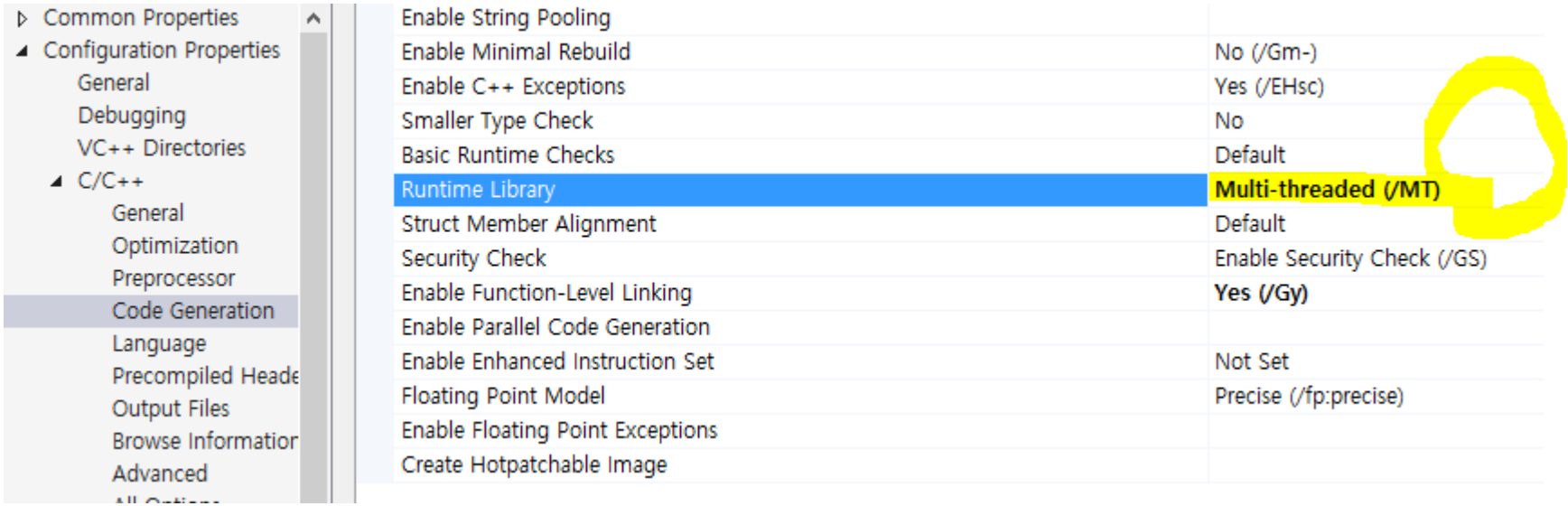
Additional Dependencies

- Linker → input



Runtime Libary

- Multi-threaded(/MT)



Basic Code Structure

```
#include <iostream>

#include "../../src/btBulletDynamicsCommon.h"

int main(void)
{
	// Code here
}
```

Headers

- #include <iostream>
- #include "../../src/btBulletDynamicsCommon.h“
    - 이 예제에서는 btBulletDynamicsCommon을 이용

시뮬레이션 환경 설정

- 바운딩 박스를 이용하여 충돌 객체를 계산하는 환경

```
btBroadphaseInterface* broadphase = new btDbvtBroadphase();
```

- 기본 충돌 설정

```
btDefaultCollisionConfiguration* collisionConfiguration = new btDefaultCollisionConfiguration();
```

- 충돌 디스패쳐

```
btCollisionDispatcher* dispatcher = new btCollisionDispatcher(collisionConfiguration);
```

- 솔버(Solver) – 객체들의 상호작용을 계산

```
btSequentialImpulseConstraintSolver* solver = new btSequentialImpulseConstraintSolver;
```

- 시뮬레이션 환경

```
btDiscreteDynamicsWorld* dynamicsWorld = new btDiscreteDynamicsWorld(dispatcher, broadphase, solver, collisionConfiguration);
```

세계 설정

- 중력 설정
- dynamicsWorld->setGravity(btVector3(0, -10, 0));

객체 외형 생성

- 객체 생성 (평면, 구)
- btCollisionShape* groundShape = new btStaticPlaneShape(btVector3(0, 1, 0), 1);

- btCollisionShape* fallShape = new btSphereShape(1);

시뮬레이션을 위한 강체 객체 생성

```
btDefaultMotionState* groundMotionState = new btDefaultMotionState(btTransform(btQuaternion(0, 0, 0, 1), btVector3(0, -1, 0)));
btRigidBody::btRigidBodyConstructionInfo
groundRigidBodyCI(0, groundMotionState, groundShape, btVector3(0, 0, 0));
btRigidBody* groundRigidBody = new btRigidBody(groundRigidBodyCI);
dynamicsWorld->addRigidBody(groundRigidBody);

btDefaultMotionState* fallMotionState =
new btDefaultMotionState(btTransform(btQuaternion(0, 0, 0, 1), btVector3(0, 50, 0)));
btScalar mass = 1;
btVector3 fallInertia(0, 0, 0);
fallShape->calculateLocalInertia(mass, fallInertia);
btRigidBody::btRigidBodyConstructionInfo fallRigidBodyCI(mass, fallMotionState, fallShape, fallInertia);
btRigidBody* fallRigidBody = new btRigidBody(fallRigidBodyCI);
dynamicsWorld->addRigidBody(fallRigidBody);
```

시뮬레이션

```
for (int i = 0; i < 300; i++) {
	dynamicsWorld->stepSimulation(1 / 60.f, 10);

	btTransform trans;
	fallRigidBody->getMotionState()->getWorldTransform(trans);

	std::cout << "sphere height: " << trans.getOrigin().getY() << std::endl;
}
```

정리

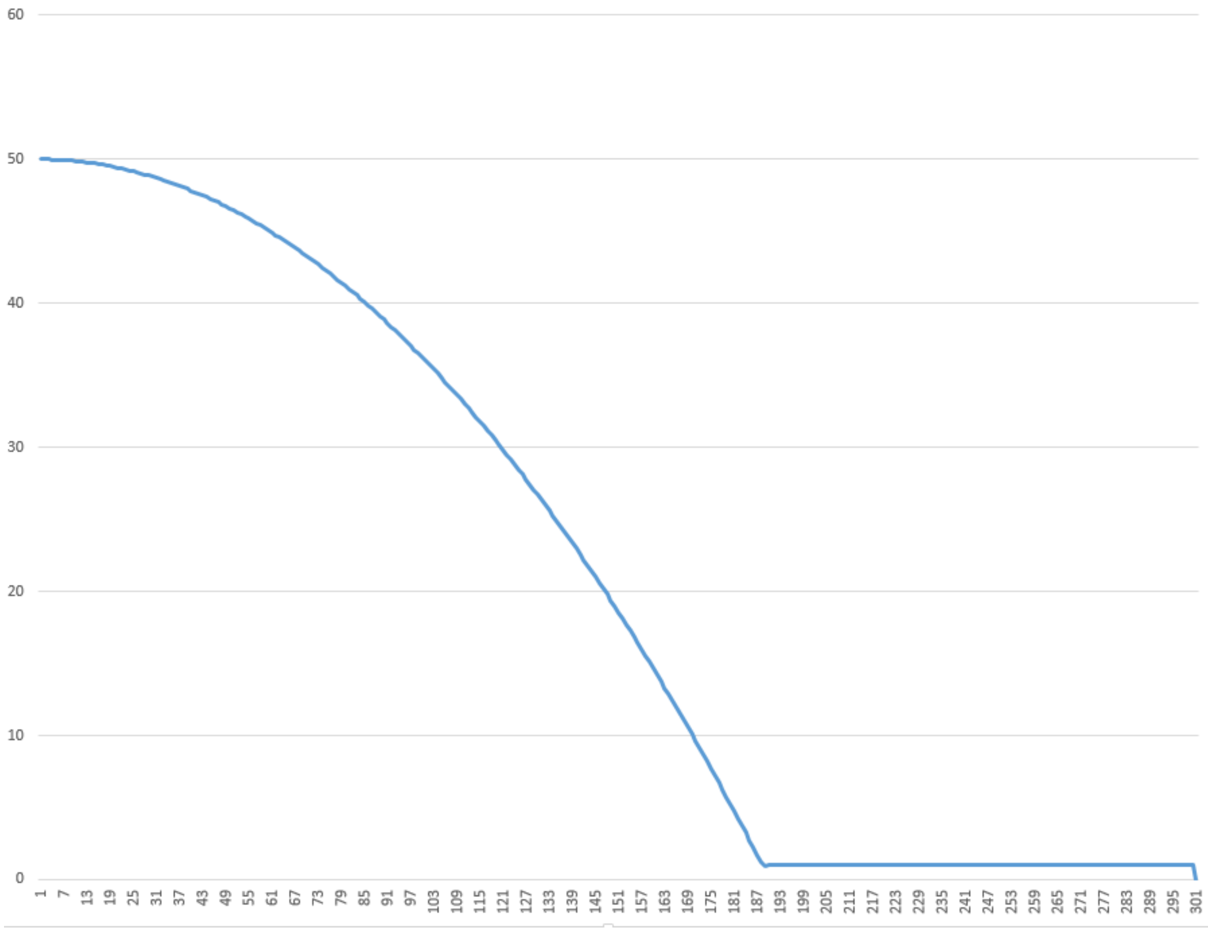
```
dynamicsWorld->removeRigidBody(fallRigidBody);
delete fallRigidBody->getMotionState();
delete fallRigidBody;

dynamicsWorld->removeRigidBody(groundRigidBody);
delete groundRigidBody->getMotionState();
delete groundRigidBody;

delete fallShape;
delete groundShape;
```

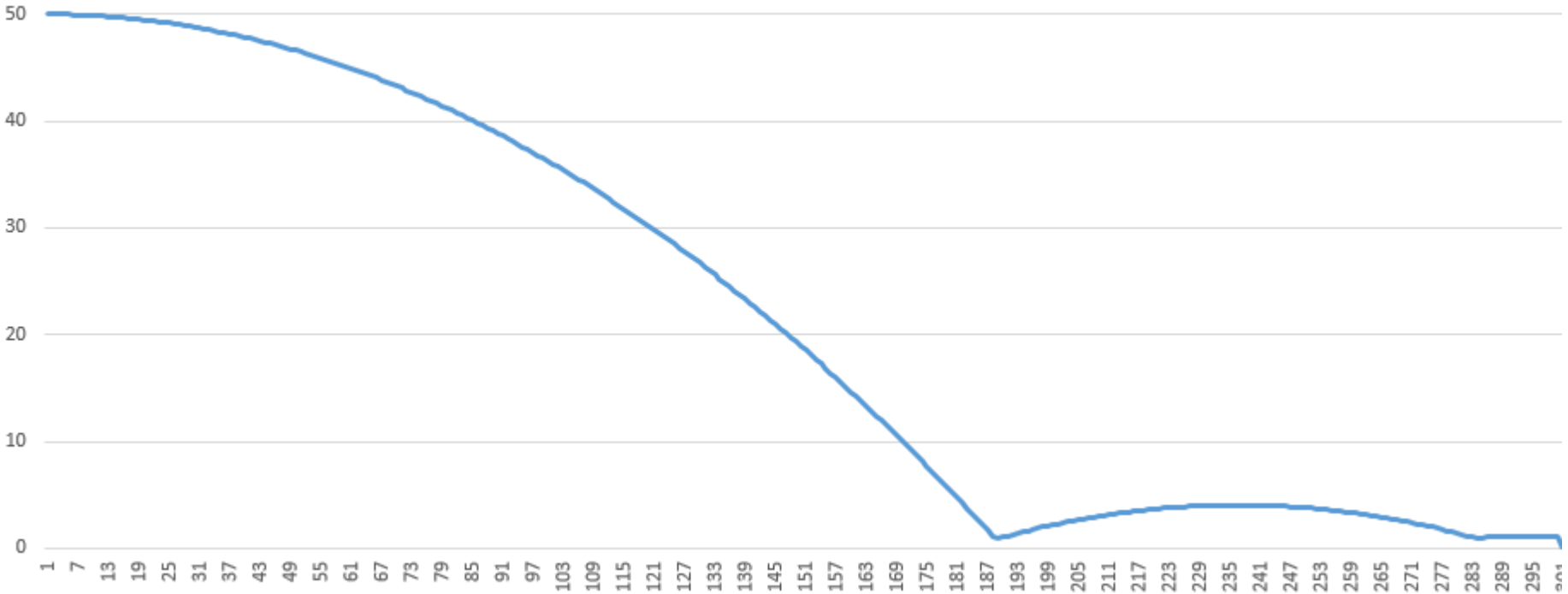
결과

- 50.000000,49.997223,49.991669,49.983334,49.972221,49.958332,49.941666,49.922222,49.900002,49.875000,49.847221,49.816666,49.783333,49.747223,49.708336,49.666668,49.622223,49.575001,49.525002,49.472225,49.416668,49.358334,49.297222,49.233334,49.166668,49.097225,49.025002,48.950001,48.872223,48.791668,48.708336,48.622223,48.533333,48.441666,48.347221,48.250000,48.150002,48.047222,47.941666,47.833332,47.722221,47.608334,47.491669,47.372223,47.250000,47.125000,46.997223,46.866669,46.733334,46.597221,46.458332,46.316666,46.172222,46.025002,45.875000,45.722221,45.566666,45.408333,45.247223,45.083336,44.916668,44.747223,44.575001,44.400002,44.222225,44.041668,43.858334,43.672222,43.483334,43.291668,43.097225,42.900002,42.700001,42.497223,42.291668,42.083336,41.872223,41.658333,41.441666,41.222221,41.000000,40.775002,40.547222,40.316666,40.083332,39.847221,39.608334,39.366665,39.122219,38.874996,38.624996,38.372219,38.116665,37.858330,37.597218,37.333328,37.066662,36.797218,36.524998,36.249996,35.972218,35.691662,35.408329,35.122219,34.833328,34.541660,34.247215,33.949993,33.649994,33.347218,33.041660,32.733326,32.422215,32.108326,31.791658,31.472214,31.149992,30.824991,30.497213,30.166658,29.833324,29.497213,29.158325,28.816658,28.472214,28.124992,27.774992,27.422215,27.066660,26.708326,26.347214,25.983326,25.616659,25.247215,24.874992,24.499992,24.122215,23.741659,23.358326,22.972216,22.583326,22.191660,21.797216,21.399994,20.999994,20.597218,20.191662,19.783329,19.372219,18.958330,18.541664,18.122219,17.699997,17.274998,16.847219,16.416664,15.983332,15.547221,15.108333,14.666666,14.222222,13.775001,13.325001,12.872224,12.416669,11.958336,11.497225,11.033337,10.566670,10.097226,9.625005,9.150005,8.672228,8.191673,7.708341,7.222230,6.733342,6.241676,5.747232,5.250010,4.750011,4.247234,3.741679,3.233346,2.722236,2.208348,1.691682,1.172238,0.930003,0.986001,0.986001,0.997200,1.005622,1.011266,1.014132,1.014220,1.011531,1.006064,1.000000,0



수정

- groundRigidBody->setRestitution(0.5);
- dynamicsWorld->addRigidBody(groundRigidBody);
- fallRigidBody->setRestitution(0.5);
- dynamicsWorld->addRigidBody(fallRigidBody);



수정

- groundRigidBody->setRestitution(1.0);
- dynamicsWorld->addRigidBody(groundRigidBody);
- fallRigidBody->setRestitution(1.0);
- dynamicsWorld->addRigidBody(fallRigidBody);