2.1 讲义：正方形轨迹的代码

```
#include <ros/ros.h>
#include <mavros_msgs/CommandBool.h>
#include <mavros_msgs/SetMode.h>
#include <mavros_msgs/State.h>
#include <geometry_msgs/PoseStamped.h>

mavros_msgs::State current_state;
void state_cb(const mavros_msgs::State::ConstPtr& msg) {
    current_state = *msg;
}


geometry_msgs::PoseStamped local_pos;
void local_pos_cb(const geometry_msgs::PoseStamped::ConstPtr& msg) {
    local_pos = *msg;
}

int main(int argc, char **argv)
{
    ros::init(argc, argv, "offb_node");
    ros::NodeHandle nh;


    ros::Subscriber local_pos_sub = nh.subscribe<geometry_msgs::PoseStamped>
        ("mavros/local_position/pose", 10, local_pos_cb);
    ros::Subscriber state_sub = nh.subscribe<mavros_msgs::State>
        ("mavros/state", 10, state_cb);
    ros::Publisher local_pos_pub = nh.advertise<geometry_msgs::PoseStamped>
        ("mavros/setpoint_position/local", 10);
    ros::ServiceClient arming_client = nh.serviceClient<mavros_msgs::CommandBool>
        ("mavros/cmd/arming");
    ros::ServiceClient set_mode_client = nh.serviceClient<mavros_msgs::SetMode>
        ("mavros/set_mode");

    ros::Rate rate(20.0);

    while (ros::ok() && current_state.connected) {
        ROS_INFO_STREAM("unconnected");
        ros::spinOnce();
        rate.sleep();
    }
```

```
geometry_msgs::PoseStamped pose;


for (int i = 100; ros::ok() && i > 0; --i) {
    local_pos_pub.publish(pose);
    ros::spinOnce();
    rate.sleep();
}

mavros_msgs::SetMode offb_set_mode;
offb_set_mode.request.custom_mode = "OFFBOARD";

mavros_msgs::CommandBool arm_cmd;
arm_cmd.request.value = true;

ros::Time last_request = ros::Time::now();

int step = 0;
int sametimes = 0;

while (ros::ok()) {
    if (current_state.mode != "OFFBOARD" &&
        (ros::Time::now() - last_request > ros::Duration(5.0))) {
        if (set_mode_client.call(offb_set_mode) &&
            offb_set_mode.response.mode_sent) {
            ROS_INFO("Offboard enabled");
        }
        last_request = ros::Time::now();
    }
    else {
        if (!current_state.armed &&
            (ros::Time::now() - last_request > ros::Duration(5.0))) {
            if (arming_client.call(arm_cmd) &&
                arm_cmd.response.success) {
                ROS_INFO("Vehicle armed");
            }
            last_request = ros::Time::now();
        }
        else
        {
            switch (step)
            {
            case 0:
                    //take off to 2m
                    pose.pose.position.x = 0;
                    pose.pose.position.y = 0;
                    pose.pose.position.z = 2;
                    //
                    if (local_pos.pose.position.z > 1.9 && local_pos.pose.position.z <
2.1)
                    {
                        if (sametimes > 100)
                        {

                            sametimes = 0;
                            step = 1;
                            pose.pose.position.x = 2;
                            pose.pose.position.y = 0;
                            pose.pose.position.z = 2;
                        }
                        else
                            sametimes++;
                    }
                    else
                    {
                        sametimes = 0;
                    }
                    local_pos_pub.publish(pose);
                    break;
                case 1:

                    if (local_pos.pose.position.x > 1.9 && local_pos.pose.position.x <
2.1)
                    {
                        if (sametimes > 100)
                        {

                            step = 2;
                            pose.pose.position.x = 2;
                            pose.pose.position.y = 2;
                            pose.pose.position.z = 2;
                        }
                        else
                            sametimes++;
                    }
                    else
                    {
                        sametimes = 0;
                    }
                    local_pos_pub.publish(pose);
                    break;
                case 2:

                    if (local_pos.pose.position.y > 1.9 && local_pos.pose.position.y < 2.1)
                    {
                        if (sametimes > 100)
                        {

                            step = 3;
                            pose.pose.position.x = 0;
                            pose.pose.position.y = 2;
                            pose.pose.position.z = 2;
                        }
                        else
                            sametimes++;
                    }
                    else
                    {
                        sametimes = 0;
                    }
                    local_pos_pub.publish(pose);
                    break;
                case 3:

                    if (local_pos.pose.position.x > -0.1 && local_pos.pose.position.x < 0.1)
                    {
                        if (sametimes > 100)
                        {

                            step = 4;
                            pose.pose.position.x = 0;
                            pose.pose.position.y = 0;
                            pose.pose.position.z = 2;
                        }
                        else
                            sametimes++;
                    }
                    else
                    {
                        sametimes = 0;
                    }
                    local_pos_pub.publish(pose);
                    break;
                case 4:

                    if (local_pos.pose.position.y > -0.1 && local_pos.pose.position.y <
0.1)
                    {
                        if (sametimes > 100)
                        {
                            step = 5;
                        }
                        else
                            sametimes++;
                    }
                    else
                    {
                        sametimes = 0;
                    }
                    local_pos_pub.publish(pose);
                    break;
                case 5:
                    offb_set_mode.request.custom_mode = "AUTO.LAND";
                    if (current_state.mode != "AUTO.LAND" && (ros::Time::now() -
last_request > ros::Duration(5.0)))
                    {

                        if (set_mode_client.call(offb_set_mode) &&
offb_set_mode.response.mode_sent)
                        {
                            ROS_INFO("AUTO.LAND enabled");
                        }
                        last_request = ros::Time::now();
                    }
                    break;
                default:
                    break;
                }
                //if (step == 5)
                    //break;
            }
        }
        ros::spinOnce();
        rate.sleep();
    }

/*
    offb_set_mode.request.custom_mode = "AUTO.LAND";
    while (ros::ok())
    {
        if (current_state.mode != "AUTO.LAND" && (ros::Time::now() - last_request > ros::Duration(5.0)))
        {

            if (set_mode_client.call(offb_set_mode) && offb_set_mode.response.mode_sent)
            {
                ROS_INFO("AUTO.LAND enabled");
            }
            last_request = ros::Time::now();
        }

        local_pos_pub.publish(pose);

        ros::spinOnce();
        rate.sleep();
    }
*/
    return 0;
}
```