A08 – Handling motion

The Vulkan application whose source code is contained in file `Assignment08.cpp`, asks you to implement the motion of a robot inside a space station. File `updatePosition.cpp` should contain the code for moving the robot. The robot must be able to move around the station, and it should always face the direction where it is going. It can float in the air, and it can move inside walls and in the outer space as a ghost.

You should modify the procedure `glm::mat4 getRobotWorldMatrix(GLFWwindow* window)` which must create and return a `glm::mat4` world matrix for the robot. It receives as input a `GLFWwindow* window` object to allow you to use the input control procedures offered by GLFW. The code already includes both the headers necessary for GLFW, as well as `<chrono>` to allow you to calculate the time since the last call in second.

You can add you own local static variables for remembering the position and direction of the robot, and you are free to choose your preferred keys and control scheme. Static variable `static glm::vec3 pos` is provided just as a sample that you could change for implementing your own version.