

```

/**
 * @file autoStack.c
 * @brief Drive forward and stack cones from the loader
 * Copyright (C) 2017 Ethan Wells
 *
 * This program is free software: you can redistribute it and/or modify it
 * under the terms of the GNU General Public License as published by the Free
 * Software Foundation, either version 3 of the License, or (at your option) any
 * later version.
 *
 * This program is distributed in the hope that it will be useful, but WITHOUT
 * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
 * FOR A PARTICULAR PURPOSE. See the GNU General Public License for more
 * details.
 *
 * You should have received a copy of the GNU General Public License along
 * with this program. If not, see <https://www.gnu.org/licenses/>
 */

#include "../include/auto.h"

void updateLinesDrive();

void autonStack() {
    update();
    driveSet(35, 35);
    int angle;

    Mutex m = angleFromUpcomingLine(&angle, NULL, 5000);
    mutexTakeDelete(m, -1);

    angle = -angle;

#ifdef DEBUG_MODE
    printf("\n\n\r%d\n\n", angle);
#endif
    sensorRefresh(&gyro);
#ifdef DEBUG_MODE
    printf("\n\n\r%d\n\n", gyro.averageVal);
#endif
    gyro.zero = angle;
    gyro.child->zero = angle;
    sensorRefresh(&gyro);
#ifdef DEBUG_MODE
    printf("\n\n\r%d\n\n", gyro.averageVal);
#endif
}

```

```

driveToPositionAngle(drivePos(0) - 300, drivePos(1) - 300, 0, 2000);

turnTo(90, 3500);

#ifdef DEBUG_MODE
    print("\n\n\rturned and stuff\n\n");
#endif

driveSet(25, 25);
while (gline(0) + gline(1) + gline(2)) {
    updateLinesDrive();
    delay(10);
}

intake.power = 50;
motorUpdate(&intake);
liftSettings.target = LIFT_3_QUARTER;
TaskHandle delet = GO(liftPID, NULL);

driveToPositionAngle(drivePos(0) + 350, drivePos(1) + 350, 90, 1600);

taskDelete(delet);
liftToPosition(LIFT_3_QUARTER, 750);

#ifdef DEBUG_MODE
    print("\n\n\rready to stack\n\n");
#endif

for (int i = 0; i < 5; i++) {
    intake.power = -75;
    update();
    delay(500);
    intake.power = -30;
    update();
    liftToPosition(LIFT_QUARTER / 2 + (LIFT_QUARTER * .25 * i), 3000);
    intake.power = 127;
    update();
    delay(400);
    intake.power = 10;
    update();
    liftToPosition(LIFT_3_QUARTER, 3000);
}
} /* autonLeftRed12 */

```