#include "MyAlgo.h"

#include <stdio.h>

#include <stdlib.h>

bool \_\_stdcall Algo1(int S, int L, int R, int Init, int& VL, int& VR){

VL = 130;

VR = 130;

return true;

}

bool \_\_stdcall Algo2(int S, int L, int R, int Init, int& VL, int& VR){

VL = 120;

VR = 120;

return true;

}

bool \_\_stdcall Algo3(int S, int L, int R, int Init, int& VL, int& VR){

VL = 120;

VR = 120;

return true;

}

bool \_\_stdcall Algo4(int S, int L, int R, int Init, int& VL, int& VR){

VL = 120;

VR = 120;

return true;

}

bool \_\_stdcall Algo5(int S, int L, int R, int Init, int& VL, int& VR){

VL = 120;

VR = 120;

return true;

}

bool \_\_stdcall Algo6(int S, int L, int R, int Init, int& VL, int& VR){

double u = PIDKontrolSistemi(S);

if (u <= -1) {

VR = 1100 + u \* 10;

VL = 1100 - u \* 10;

}

else if (u >= 1) {

VR = 1100 + u \* 10;

VL = 1100 - u \* 10;

}

else {

VR = 1100 + u \* 10;

VL = 1100 + u \* 10;

}

if (VR > 127)

VR = 60;

else if (VR < 127)

VR = 5;

if (VL > 127)

VL = 60;

else if (VL < 127)

VL = 5;

return true;

}

bool \_\_stdcall Algo7(int S, int L, int R, int Init, int& VL, int& VR){

VL = 120;

VR = 120;

return true;

}

float PIDKontrolSistemi(int S) {

double u, integral, öncekihata, hata, türev, dt = 1;

double a = 992;//(decimal degeri onluk sayı sistemine çevirdim.)

double Kp = 1 , Ki = 0 , Kd = 0;//(Kp degeriyle robotun sallanması ve yalpalanması ayarlanır. Ki , Kd daha ince ayarlar. Kp den sonra Kd sonra Ki ayarlanabilir.)

öncekihata = 0;

integral = 0;

hata = S - a; //(Hedef-decimal deger)

integral = integral + (hata\*dt); //( ∫ e\*dt)

türev = (hata - öncekihata) / dt;//(de/dt)

u = (Kp\*hata) + (Ki\*integral) + (Kd\*türev);//u=[(Kp\*e)+(Ki ∫ e\*dt)+(Kd\*(de/dt))](Robotun hareketini saglayan (oransal)+(integral)+(türev))

öncekihata = hata;

return u;

}

#ifndef \_\_MAIN\_H\_\_

#define \_\_MAIN\_H\_\_

#include <windows.h>

#ifdef BUILD\_DLL

#define DLL\_EXPORT \_\_declspec(dllexport)

#else

#define DLL\_EXPORT \_\_declspec(dllimport)

#endif

#ifdef \_\_cplusplus

#endif

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo1(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo2(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo3(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo4(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo5(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo6(int S, int L, int R, int Init, int& VL, int& VR);

extern "C" \_\_declspec(dllexport) bool \_\_stdcall Algo7(int S, int L, int R, int Init, int& VL, int& VR);

static float PIDKontrolSistemi(int S);

static float u = 0.0f;

#ifdef \_\_cplusplus

#endif

#endif