Ödev II :

Hazırlanıp teslim edilecekler:

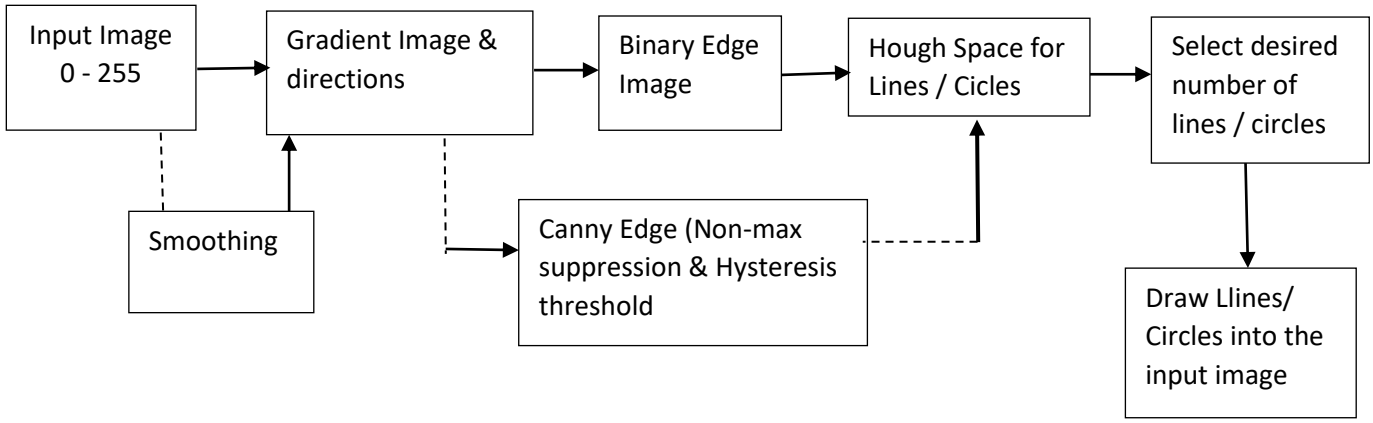
1) C/C++ programlama dilinde yazılan fonksiyonların sadece ham kodları, **2)** Her bir fonksiyonu giriş çıkış değerlerini (resim v.b.) ve süreçleri kısaca açıklayan raporları pdf dosya olarak hazırlanıp en **son teslimi tarihi : 31 Mayıs 2023 Çarşamba, Saat 23.55** : e kadar sisteme yüklenmelidir.

2) Ödevlerin her ikisinin (Ödev-1 ve Ödev –2) sunumları **5-9 Haziran 2023** tarihlerinde olacaktır. Ödev rapor ve kodları tesliminde 5-9 Haziran 2023 tarihlerinde hangi saat aralığında olabileceği önerisinde bulunuz. Ödev sunumunda her bir kişi kendine ayrılan maksimum 30 dakika için Zoom dan oturum açıp linki sunum saati öncesi ders sorumlusu e-mail adresine (ekinci@ktu.edu.tr)

Teslim edilecek ödev-II için en temel seviyede yapılması istenen süreçler ve ham C/C++ kodları için çağırma ve prototype fonksiyonları proje çalışması, teslimi ve sunumu için öğrencilere örnek yapılar olarak aşağıda verilmiştir.

Canlı sunumlarda ödevde yapılan her bir fonksiyonun giriş ve çıkış verilerini gösterecek şekilde ara yüz yapısı kurulmalıdır. Aşağıda verilen süreçler istenen minimum süreçlerdir. İlgili derslerde hedef çıktılara ulaşılması için olası tüm işlev ve süreçlerin detayları teorik ve uygulamalı açıklanmıştır.

Ödev II içeriği : Görüntüdeki istenen adetteki Doğrusal (Line) ve Dairesel (Circle) yapıdaki sınırların (kenarlar) ve nesnelerin konumlarının bulunması




```
int* EdgeImage = Gradient(raw_intensity, Width, Height, angles);

int* Gradient(Byte* raw_intensity, int  Width, int Height, double&* angles);

Byte* BinaryEdgeImage = EdgeBinarized(EdgeImage, Width, Height);
```

Byte* EdgeBinarized(Byte* EdgeImage, int Width, int Height);

int* HoughLine = HoughSpaceForLine = Hough_Line(Byte* BinaryEdgeImage, int Width, int Height, int * angle, int& HoughWidth, int& HoughHeight);

int* HoughCircle = HoughSpaceForCircle = Hough_Line(Byte* BinaryEdgeImage, int Width, int Height, int * angle, int& HoughWidth, int& HoughHeight, int radius);

DrawSelectedLines(int* HoughLine, int HoughWidth, int HoughHeight, int* raw_intensity, int Width, int Height, int number_of_lines);

DrawSelectedCircles(int* HoughLine, int HoughWidth, int HoughHeight, int* raw_intensity, int Width, int Height, int number_of_circles);

---- Canny Edge işlevi kullanılarak BinaryEdgeImage ürerilmek istenirse

Byte* Smoothing(raw_intensity, With, Height);

Byte* smooth = Smoothing(raw_intensity, With, Height);

int* EdgeImage = Gradient(smooth, Width, Height, angles);

int* Gradient(Byte* smooth, int Width, int Height, double&* angles);

Byte* CannyEdge(EdgeImage, Width, Height, angles);

Byte* BinaryEdgeImage = CannyEdge(Byte* EdgeImage, int Width, int Height, double* angles);