**YILDIZ TEKNİK ÜNİVERSİTESİ**

**ELEKTRİK – ELEKTRONİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**BLM2021 – ALT SEVİYE PROGRAMLAMA**

**ÖDEV – 2**

Öğrenci No: 21011055

Ad – Soyad: Emirhan Özsaray

Telefon No: 05418493876

E-Posta Adresi: [emirhan.ozsaray@std.yildiz.edu.tr](mailto:emirhan.ozsaray@std.yildiz.edu.tr)

Öğretim Görevlisi: Dr. Öğr. Üyesi Erkan Uslu

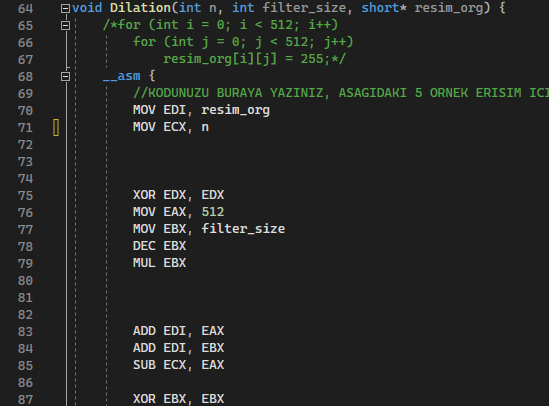
# İÇİNDEKİLER

1) Dilation Fonksiyonu Anlatımı

2) Erosion Fonksiyonu Anlatımı

3) Kaynakça

1) Dilation Fonksiyonu Anlatımı



1) EDI registerına resim\_org dizisinin başlangıç adresi atanır. ECX registerına n, yani bu dizinin uzunluğu atanır.

2) EDX sıfırlanır. EDX’te döngünün kaç kere döndüğü korunur. EAX registerına 512 sayısı yani resmin bir satırının uzunluğu atanır. EBX registerına filter\_size atanıp 1 eksiltilir. EAX ile çarpılır. EDI registerına EAX ve EBX eklenir. Bunu yapmamın sebebi filtrenin adres dışına taşmamasını sağlayıp filtrenin sadece dizi içinde kalmasını sağlamaktır. EAX geçilecek satırı belirlerken EBX geçicelek sütunu belirler. ECX’ten EAX çıkarılarak filte boyutuna göre loopun son satırlara girmesi engellenir.

3) EBX sıfırlanır. Bunu yapmamın sebebi dizide gezmek için EBX’i kullanacak olmamdır. [EDI+EBX] olarak kullanacağım için dizinin ilk adresini kaybetmememi sağlar.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

4) EAX max değeri tutacağım registerdır. O yüzden her döngü başında 0’a eşitlenir.

5) ECX stacke pushlanır. L2 loopuna girmeden önce değer kaybolmasın diye pushlanır ve filtrenin büyüklüğü ECX’e atanır. L2 ve L3 loopları filteyi gezeceği için loopların tur sayısı filtre büyüklüğü kadar olmalıdır.

6) 96-108 arası satırlarda ilk önce EBX’in hangi elemanı gösterdiği saklansın diye pushlanır. Sonrasında EBX filtrenin en sol üstündeki elemanı gösterecek şekilde ayarlanır. Örnek olarak filter\_size 5 girilirse ECX değeri 4 olur EAX ile çarpılıp 2048 değeri elde edilir. ECX 1’e tekrar 1 eklenir. EBX’ten EAX yani 2048 ve ECX yani 5 çıkartılır. Sonra EBX 1 arttırılır. Böylece 5 büyüklüğündeki bir filtrenin sol üstü gösterilir.

metin, ekran, siyah, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

7) L2 loopunda ECX değeri korunması gerektiğinden yine pushlanır ve filter\_size tekrar atanır.

8) L3 loopunda filtrenin gösterdiği dizi değeri max değerle karşılaştırılır. Eğer max değerden büyükse AX’e atanır. Değilse BRANCH’e atlayıp devam eder. Filtreyi soldan sağa gezmek için EBX’e 2 eklenir ve sonraki elemana geçer.

9) EBX değeri stackten alınır ve 1024 eklenir. Bu şekilde filtrede sonraki satıra geçilmiş olunur.

10) L2 loopu bittikten sonra EBX stackten poplanıp dizi elemanında gezme işine geçer.

11) ECX poplanır.

12) AX’teki max değer stacke atılır.

13) EDX 1 arttırılarak counter görevi görür.

metin, siyah, ekran içeren bir resim

Açıklama otomatik olarak oluşturuldu

14) EBX stacke atılır ve filter\_size’ a eşitlenir.

15) EAX’e 513 değeri atılır. EBX çıkartılır. EBX’in eski değeri poplanır.

16) EDX yani counter ile EAX karşılaştırılır. Eğer eşitlerse EBX’e filter\_size\*2-2 değeri eklenir. Bu kısım filtrenin kenarlardaki elemanlara gitmemesi içindir. Eğer bir sonraki adım kenardaki bir elemana gidecekse EBX’e sonraki satıra geçmesi için yapılır. EAX SHR ile yarıya düşürülür ve ECX’ten çıkartılır. Bunun nedeni atladığımız elemanlar için ECX’te hala değer olmasıdır. Bu elemanlar atlandığı için atlanan eleman kadar ECX azaltılır. Counter yani EDX sıfırlanır. EBX’e 2 eklenir. Bu şekilde eğer counter eşitlik yoksa EBX bir sonraki elemana geçer. Eşitlik varsa EBX filtrenin tamamının dizi içinde kalması için bir sonraki elemana geçer.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

17) 170-176 arası EBX in arrayin filtre boyutuna göre sonuna atanması ve ECX’ten dizinin ilk satırına geçmemesi için filtre boyutuna göre EAX çıkartılır. (L1’in sonunda EAX’te zaten filtre boyutunun 2 katının 2 eksiğinin yarısı bulunduğu için EBX’ten 2 çıkartıp sonra EAX’in iki katını çıkarttım.) EDX yani counter 0 lanır.

18) Stacke atılan max değerler teker teker poplanır ve diziye sondan başa doğru atanmaya başlar.

19) 184-200 arası L1’in sonunda olan yerle nerdeyse aynı tek fark olarak EBX’e filter\_size göre ekleme yapmak yerine ve 2 eklemek yerine, aynı sayıları çıkartıyorum ki geri geri gitsin.

iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldumetin, kıyafet, iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldukişi, şapka içeren bir resim

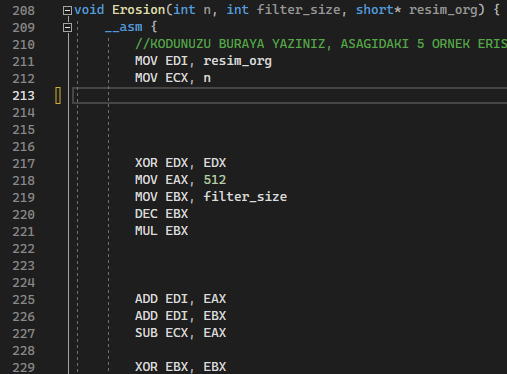
Açıklama otomatik olarak oluşturuldu

3x3 filtre

5x5 filtre

7x7 filtre

2) Erosion Fonksiyonu Anlatımı



1) EDI registerına resim\_org dizisinin başlangıç adresi atanır. ECX registerına n, yani bu dizinin uzunluğu atanır.

2) EDX sıfırlanır. EDX’te döngünün kaç kere döndüğü korunur. EAX registerına 512 sayısı yani resmin bir satırının uzunluğu atanır. EBX registerına filter\_size atanıp 1 eksiltilir. EAX ile çarpılır. EDI registerına EAX ve EBX eklenir. Bunu yapmamın sebebi filtrenin adres dışına taşmamasını sağlayıp filtrenin sadece dizi içinde kalmasını sağlamaktır. EAX geçilecek satırı belirlerken EBX geçicelek sütunu belirler. ECX’ten EAX çıkarılarak filte boyutuna göre loopun son satırlara girmesi engellenir.

3) EBX sıfırlanır. Bunu yapmamın sebebi dizide gezmek için EBX’i kullanacak olmamdır. [EDI+EBX] olarak kullanacağım için dizinin ilk adresini kaybetmememi sağlar.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

4) EAX min değeri tutacağım registerdır. O yüzden her döngü başında 255’e eşitlenir.

5) ECX stacke pushlanır. L2 loopuna girmeden önce değer kaybolmasın diye pushlanır ve filtrenin büyüklüğü ECX’e atanır. L2 ve L3 loopları filteyi gezeceği için loopların tur sayısı filtre büyüklüğü kadar olmalıdır.

6) 238-250 arası satırlarda ilk önce EBX’in hangi elemanı gösterdiği saklansın diye pushlanır. Sonrasında EBX filtrenin en sol üstündeki elemanı gösterecek şekilde ayarlanır. Örnek olarak filter\_size 5 girilirse ECX değeri 4 olur EAX ile çarpılıp 2048 değeri elde edilir. ECX 1’e tekrar 1 eklenir. EBX’ten EAX yani 2048 ve ECX yani 5 çıkartılır. Sonra EBX 1 arttırılır. Böylece 5 büyüklüğündeki bir filtrenin sol üstü gösterilir.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

7) L2 loopunda ECX değeri korunması gerektiğinden yine pushlanır ve filter\_size tekrar atanır.

8) L3 loopunda filtrenin gösterdiği dizi değeri min değerle karşılaştırılır. Eğer min değerden küçükse AX’e atanır. Değilse BRANCH’e atlayıp devam eder. Filtreyi soldan sağa gezmek için EBX’e 2 eklenir ve sonraki elemana geçer.

9) EBX değeri stackten alınır ve 1024 eklenir. Bu şekilde filtrede sonraki satıra geçilmiş olunur.

10) L2 loopu bittikten sonra EBX stackten poplanıp dizi elemanında gezme işine geçer.

11) ECX poplanır.

12) AX’teki min değer stacke atılır.

13) EDX 1 arttırılarak counter görevi görür.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

14) EBX stacke atılır ve filter\_size’ a eşitlenir.

15) EAX’e 513 değeri atılır. EBX çıkartılır. EBX’in eski değeri poplanır.

16) EDX yani counter ile EAX karşılaştırılır. Eğer eşitlerse EBX’e filter\_size\*2-2 değeri eklenir. Bu kısım filtrenin kenarlardaki elemanlara gitmemesi içindir. Eğer bir sonraki adım kenardaki bir elemana gidecekse EBX’e sonraki satıra geçmesi için yapılır. EAX SHR ile yarıya düşürülür ve ECX’ten çıkartılır. Bunun nedeni atladığımız elemanlar için ECX’te hala değer olmasıdır. Bu elemanlar atlandığı için atlanan eleman kadar ECX azaltılır. Counter yani EDX sıfırlanır. EBX’e 2 eklenir. Bu şekilde eğer counter eşitlik yoksa EBX bir sonraki elemana geçer. Eşitlik varsa EBX filtrenin tamamının dizi içinde kalması için bir sonraki elemana geçer.

metin, ekran, siyah içeren bir resim

Açıklama otomatik olarak oluşturuldu

17) 311-318 arası EBX in arrayin filtre boyutuna göre sonuna atanması ve ECX’ten dizinin ilk satırına geçmemesi için filtre boyutuna göre EAX çıkartılır. (L1’in sonunda EAX’te zaten filtre boyutunun 2 katının 2 eksiğinin yarısı bulunduğu için EBX’ten 2 çıkartıp sonra EAX’in iki katını çıkarttım.) EDX yani counter 0 lanır.

18) Stacke atılan min değerler teker teker poplanır ve diziye sondan başa doğru atanmaya başlar.

19) 325-341 arası L1’in sonunda olan yerle nerdeyse aynı tek fark olarak EBX’e filter\_size göre ekleme yapmak yerine ve 2 eklemek yerine, aynı sayıları çıkartıyorum ki geri geri gitsin.

kıyafet, kişi, kadın, iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldu

3x3 filte

metin, kıyafet, kadın, kişi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, kıyafet, kadın, atkı içeren bir resim

Açıklama otomatik olarak oluşturuldu

5x5 filtre

7x7 filtre

3) Kaynakça

<https://www.ncbi.nlm.nih.gov/books/NBK546156/box/ch3.box16/?report=objectonly>