

**BLM2021 Alt Seviye Programlama Dersi 2. Ödev Raporu**

**İsim Soyisim:** Berkay Gümüşay

**Numara:** 21011084

**Tel No:** 0 535 840 79 78

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

**Not : Erosion** ve **Dilation** 3 satır hariç tamamen aynı kodlara sahip olduğu için tek kod üzerinden anlatılmış, gereken yerlerde ayrım yapılmıştır.

**1- Satır ve sütunları gezecek döngünün adım sayısını belirleme**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. Burada **mov eax, 512** komutu ile satır sayısı olan **512**, **eax** registerine aktarılır.
2. **sub eax,** **filter\_size** ve **inc eax** komutları ile satırda ilerlenebilecek en son nokta tespit edilir.
3. **mov ecx, eax** ile döngü oluşması için bu sayı **ecx** e aktarılır.
4. **mov edx, eax** ile bu değer **edx** değişkeninde saklanır.
5. **xor eax, eax** komutu ile daha sonra kullanılacak olan **eax** registeri sıfırlanır.

**2- Resimin satırlarını gezecek olan döngü (İlk kısım)**

metin, ekran içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **resimSatirLoop:** komutu ile resmin satırlarını gezecek olan döngü oluşturulur.
2. **push ecx** ile önceki döngünün adım sayısı değişmemesi için **ecx** registeri **stack’e** atılır.
3. **mov edi, tmp** komutu ile ek dizinin adresi **edi** registerine aktarılır.
4. **mov esi, resim\_org** ile resim dizisinin adresi **esi** registerine aktarılır.
5. **add esi, eax** ve **add edi, eax** komutları ile her döngünün başında **esi** ve **edi** sonraki satıra geçer **(Devamı raporun ilerisinde)**.
6. **mov ecx, edx** ile önceden **edx’e** atadığımız adım sayısını yeni döngü için de uygularız çünkü elimizdeki resim, kare bir resimdir.
7. **push edx** ile daha sonra **dx** registerini kullanacağımız için **edx** registeri değişmesin diye **stack’e** atarız.
8. **xor edx, edx** ile **edx** registeri sıfırlanır.

**3- Resimin sütunlarını gezecek olan döngü (İlk kısım)**

**Erotion: Dilation:**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **resimSutunLoop:** komutu ile resmin sütunlarını gezecek olan döngü oluşturulur.
2. **push ecx** ile önceki döngünün adım sayısı değişmemesi için **ecx** registeri **stack’e** atılır.
3. **mov ecx, filter\_size** ile sonraki döngüde adım sayısı filtre boyutu kadar olacağı için bu değer **ecx’e** atılır.
4. **push** **eax** ile ileride **eax** üzerinde yapılacak değişikliklerden dolayı önceki değeri saklamak için **eax**, **stack’e** atılır.
5. **xor** **eax,** **eax** ile **eax** registeri sıfırlanır.
6. **mov dx, 255** ve **xor dx, dx** :

**f.1-** **Erosionda** **en küçük** arandığı için tüm değerlerden büyük olacak olan **255** değeri seçilmiştir.

**f.2-** **Dilation** fonksiyonunda ise bu değer **0** dır. Çünkü en büyük değer aranmaktadır.

**4- Filtrenin satırlarını gezecek olan döngü (İlk kısım)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **filtreSatirLoop:** komutu ile filtrenin satırlarını gezecek olan döngü oluşturulur.
2. **mov ebx, esi** komutu **resim\_org** dizisindeki mevcut değerin adresini**(filtrenin başlangıç noktası)** **ebx** registerine koyar.
3. **add ebx,eax** komutu ile resim dizisini gezerken yapıldığı gibi eax registerine ileride eklenecek değerler ile **ebx** registeri 2. Döngü ve sonrasında aşağı satırlara iner.
4. **push ecx** komutu ile döngünün adım sayısı **stackte** saklanır.
5. **mov ecx, filter\_size** ile sütunları gezecek döngünü adım sayısı olan filtre boyutu **ecx** registerine aktarılır.

**5- Filtrenin sütunlarını gezecek olan döngü (İlk kısım)**

**Erosion:**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Dilation:**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **filtreSutunLoop:** komutu ile filtrenin sütunlarını gezecek olan döngü oluşturulur.
2. **cmp dx, WORD PTR[ebx]** komutu ile filtrede gezerken mevcut değer **Erosion için Min** değerini **Dilation için ise Max** Değerini tutan **dx** registeri ile karşılaştırılır.
3. **jb kucuk (Erosion**) **/** **jae buyuk (Dilation)** komutları ile fonksiyona göre **büyüklük/küçüklük kontrolü** yapılır ve duruma göre **jump** işlemi gerçekleştirilir.
4. **mov dx, WORD PTR[ebx**] komutu ile **Erosion** **için** eğer mevcut değer **dx ten daha küçükse** **dx** mevcut değer ile güncellenir, **Dilation için** ise eğer mevcut değer **dx’ten daha büyükse** **dx** mevcut değer ile güncellenir.
5. **kucuk: add ebx, 2 /** **buyuk: add ebx, 2** komutları ile **ebx** bir sonraki değere geçer.

**6- Filtenin satırlarını gezecek olan döngü (Son Kısım)**



1. **pop ecx** ile ecx stackten çekilerek döngünün devamı sağlanır.
2. **add eax, 1024** ile öncesinde anlatıldığı gibi her adımda **ebx’in** bir satır aşağıya inmesi sağlanır.
3. **loop filtreSatirLoop** ile döngü bitirilir.

**7- Resimin sütunlarını gezecek olan döngü (Son Kısım)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **mov WORD PTR[edi], dx** ile ek dizinin mevcut noktasına filtreden elde ettiğimiz piksel yerleştirilir.
2. **pop eax/pop ecx** ile stackteki **eax** ve **ecx** değerleri **stackten** çekilir.
3. **add edi, 2**/**add esi, 2** ile **edi** ve **esi** registerleri bir sonraki değişkeni gösterecek şekilde ayarlanır.
4. **loop** **resimSutunLoop** ile loop sonlandırılır.

**8- Resimin satırlarını gezecek olan döngü (Son Kısım)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **add eax, 1024** ile her adımda **esi** ve **edi’nin** bir satır aşağı inmesi sağlanır.
2. **pop edx** ve **pop ecx** ile daha önce **stack**’e atılan **edx** ve **ecx** registerları **stackten** çekilir.
3. **loop resimSatirLoop** ile loop sonlandırılır.

**9- Ek diziyi ana resim dizisine kopyalama**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **mov ebx,tmp/mov esi, resim\_org** komutları ile **ebx** registerine **tmp** dizisinin, **esi** registerine ise **resim\_org** dizisinin adresi atanır.
2. **mov ecx, n** komutu döngünün adım sayısını **ecx** registerine atar.
3. **loop1: mov ax,[ebx]** komutu **tmp** dizisinin mevcut indexindeki değeri **ax’e** atar.
4. **mov [esi], ax** komutu ile **resim\_org** dizisinin mevcut indexine **ax** registerindeki değer yazılır.
5. **add esi,2/add ebx,2** komutu **ebx** ve **esi** registerlerini bir sonraki değere geçirir.
6. **loop loop1** komutu döngüyü sonlandırır

**8- Kodun Çalıştırılması Sonucu Üretilen Resimler**

1. **Erosion (filter\_size = 3)**

metin, kişi, kadın, iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **Erosion (filter\_size = 5)**

metin, kıyafet, kadın, kişi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **Erosion (filter\_size = 7)**

metin, kıyafet, kadın, kişi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **Dilation (filter\_size = 3)**

kişi, şapka içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **Dilation (filter\_size = 5)**

iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. **Dilation (filter\_size = 7)**

iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldu