

## C/C++ DERS KONULARI

1. Problem çözme ve algoritmalar
2. Veri Tipleri
3. Döngüler ve Kontrol komutları
4. Diziler ve Karakter dizileri
5. Pointers (İşaretçiler)
6. Fonksiyonlar
7. Structure & Union
8. Dosya Giriş/Çıkış Fonksiyonları
9. Matematik ve zaman fonksiyonları
10. Grafik Programları
11. C++ Giriş, Sınıflar

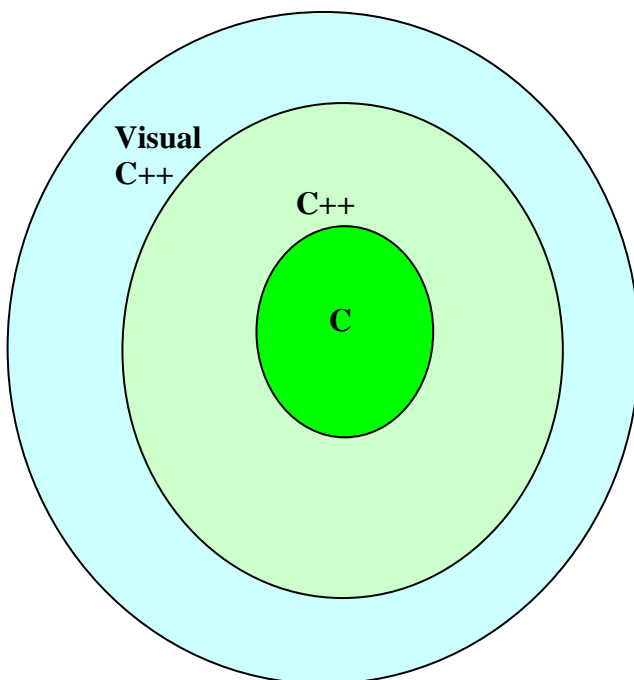
### **Kaynaklar**

1. C Programlama dili, “Kemal Yarcı”
2. C/C++, “Gürcan Banger”
3. C ve Siz, “Fatih Ekici”

### **Değerlendirme**

Yıl Sonu Notu = Vize %40 + Quiz %10 + Final %50 olarak hesaplanacaktır.

### **I. C/C++ Programlama dilinin Yeri**



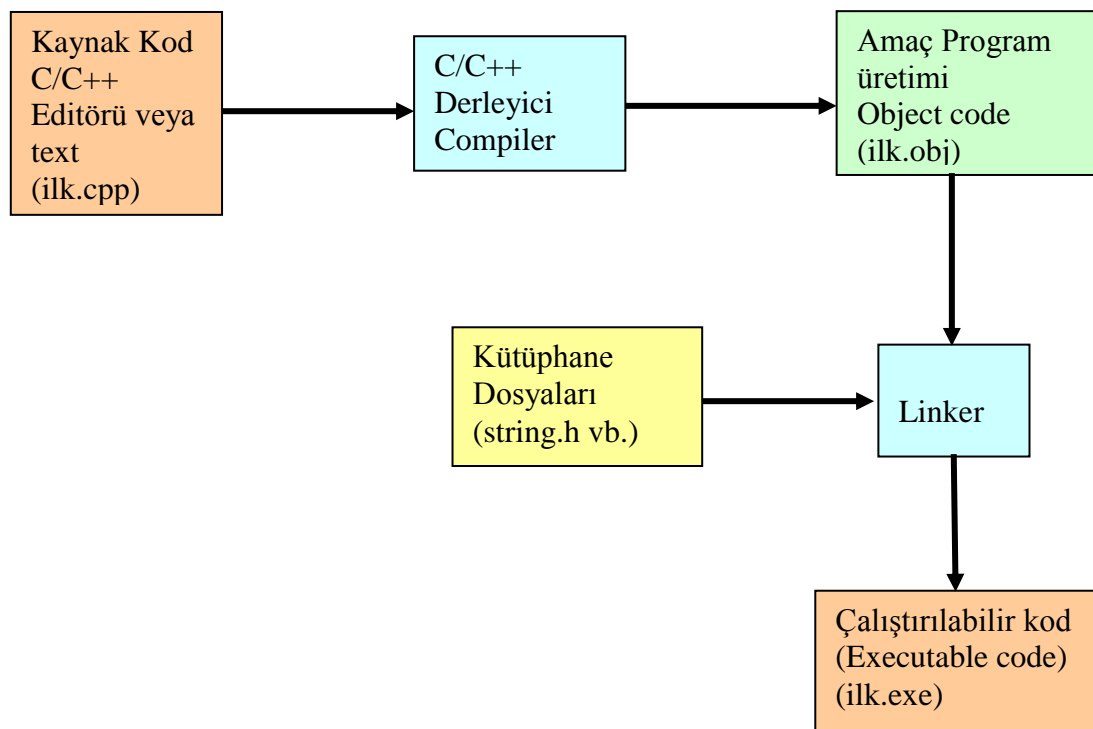
## II. C Programlama Dilinin Avantajları

- **Az Komut Kümesi** : Anahtar kelime sayısı azdır.
- **Modüler Uyumluluk** : Komutlar az yer kaplar diğer dillerde olduğundan daha az yazılım kuralı vardır. Yüksek seviyeli dillerde yaygın olan fonksiyonların bir çoğu C’de yerleşik olarak bulunmaz. Bu fonksiyonlar kütüphane dosyaları ile programlara eklenir. Aynı zamanda derleme ve bağlama işlemlerinin ayrı ayrı yapılmasının sağlar. Böylelikle birden fazla program birleştirilebilir.
- **Verim**: Derleyici bir komut veya komut grubunun tekrarladığını fark ettiğinde, yalnızca ilk blok makine koduna çevrilir, sonraki bloklar ilk kodu referans alır. Ayrıyetten C Derleyiciler erişilemeyen kod (unreachable code) ve tanımlanan ve hiç kullanılmayan değişkenleri gösterirler.
- **Tip Tanımlamasız Bir Dildir** : Değişken tipi tanımlanmasında keskin kurallar yoktur. Karakter tanımlı bir değişken tamsayı olarak kullanılabilir.
- **Yapısallık** : Programcının probleme odaklanmasının sağlayan modern dillerden beklenen tüm kontrol yapılarının içerir. Program mantıksal bütünlük gösteren bloklara bölünebilir.
- **Simgesel Dil Kullanımı ve Bit bazında işlem Yapma**: C dilinde (inline) fonksiyonu ile makine dili komutları kullanılabilir. Aynı zamanda Verilere bit seviyesinde (union) ulaşılp işlem yapma özelliği mevcuttur.
- **İşaretçi (Pointer)** : Belleğin özel bir bölümü adreslenebilir, içeriği okunup değiştirilebilir. Aritmetik işlemler yapılabilir.

### III. C Programlama Dilinin Dezavantajları

- **Güçlü bir tanımlamalı dil değildir.:** Kayan noktalı bir sayı (14.687) gibi, tam sayı olarak okunabilir, (14 olarak) işlem görebilir bu da veri kaybına neden olur. Bu programcının sorumluluğundadır.
- **Uygulama Zamanında (Run time) Denetim Eksikliği :** Derleme ve bağlama aşamasında hata vermeyen program çalışmayabilir veya yanlış çalışabilir. Bitmeyen döngü yapısı gibi. Buda programcının sorumluluğundadır.

### IV. C/C++ Programının Oluşturulma Aşamaları



## 1. Problem Çözme ve Algoritmalar

### 1.1 Problem Çözme

Problem çözmeye, soruna hemen girişmek yerine, dikkatli ve sistematik yaklaşım ilke olmalıdır. Problem iyice anlaşılmalı ve mümkün olduğu kadar küçük parçalara ayrılmalıdır. Descartes tarafından "Discourse on Method" isimli kitabında anlatılan problem çözme teknikleri;

1. Doğruluğu kesin olarak kanıtlanmadıkça, hiçbir şeyi doğru olarak kabul etmeyin; tahmin ve önyargılardan kaçının.
2. Karşılaştığınız her güçlüğü mümkün olduğu kadar çok parçaya bölün.
3. Düzenli bir biçimde düşünün; anlaşılması en kolay olan şeylerle başlayıp yavaş yavaş daha zor ve karmaşık olanlara doğru ilerleyiniz.
4. Olaya bakışınız çok genel, hazırladığınız ayrıntılı liste ise hiçbir şeyi dışarıda bırakmayacak kadar kusursuz ve eksiksiz olsun.

### 1.2 Algoritmalar

Belirli bir görevi yerine getiren sonlu sayıdaki işlemler dizisidir.

*İ.S. 9.yy da İranlı Musaoğlu Horzumlu Mehmet (Alharezmi adını araplar takmıştır) problemlerin çözümü için genel kurallar oluşturdu. Algoritma Alharezmi'nin Latince okunuşu.*

Her algoritma aşağıdaki kriterleri sağlamalıdır.

1. **Girdi:** Sıfır veya daha fazla değer dışarıdan verilmeli.
2. **Çıktı:** En azından bir değer üretilmeli.
3. **Açıklık:** Her işlem (komut) açık olmalı ve farklı anlamlar içermemeli.
4. **Sonluluk:** Her türlü olasılık için algoritma sonlu adımda bitmeli.
5. **Etkinlik:** Her komut kişinin kalem ve kağıt ile yürütebileceği kadar basit olmalıdır.

Not: İşletim sistemleri gibi program enerjisi olduğu sürece çalışır.

**Örnek :** 1'den 100'e kadar olan sayıların toplamını veren algoritma.

1. Toplam T, sayılar da i diye çağırılsın.
2. Başlangıçta T'nin değeri 0 ve i'nin değeri 1 olsun.
3. i'nin değerini T'ye ekle.

4. i'nin deęerini 1 arttır.
5. Eęer i'nin deęeri 100'den byk deęil ise 3. adıma git.
6. T'nin deęerini yaz.

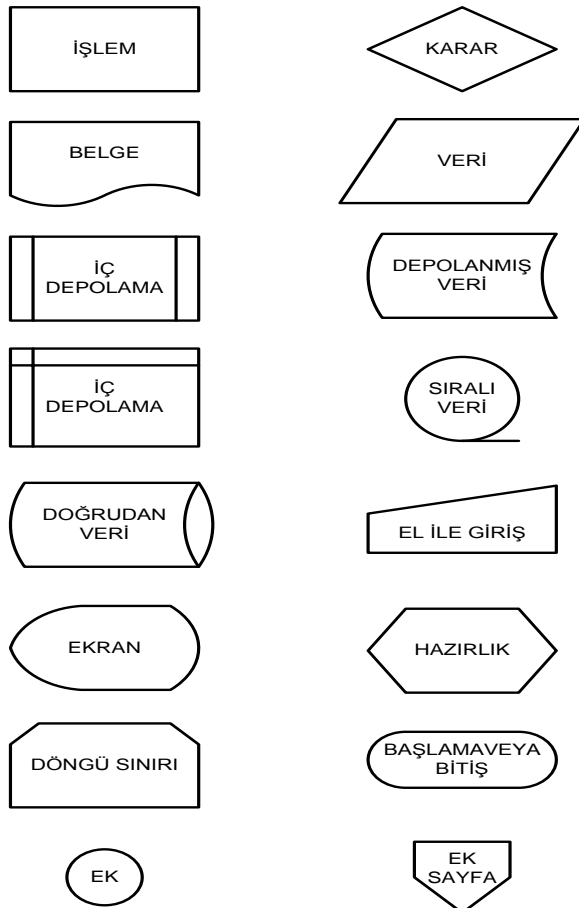
Algoritmaların yazım dili deęişik olabilir. Gnlk konuřma diline yakın bir dil olabileceęi gibi simgelere dayalı da olabilir. Akıř řeması eskiden beri kullanıla gelen bir yapıdır.

Algoritmayı yazarken farklı anlamlar taşıyan deęişik řekildeki kutulardan yararlanılır. Yine aynı amaç için kullanılan programlama diline yakın bir (szde kod = pseudo code) dil , bu kendimize zg de olabilir, kullanılabilir.[Hseyin Pektař]

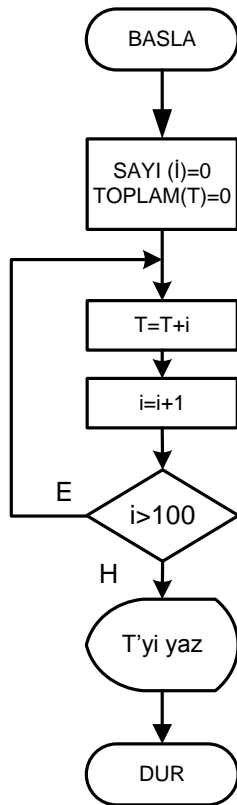
Aynı algoritmayı ařaęıdaki gibi yazabiliriz.

1. T=0 ve i=0
2. i'nin deęerini T'ye ekle.
3. i'yi 1 arttır.
4.  $i < 101$  ise 2.adıma git.
5. T'nin deęerini yaz.

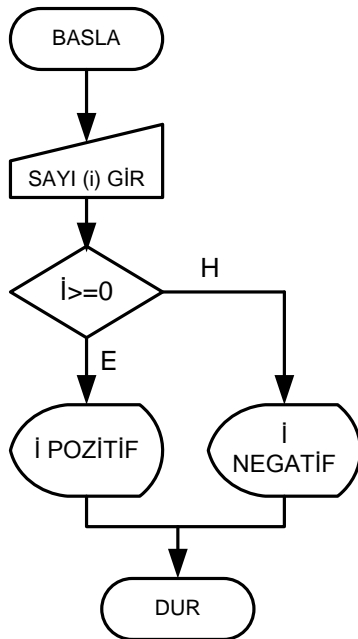
## 1.2. İřaret Akıř Diyagramında Kullanılan Bazı Semboller



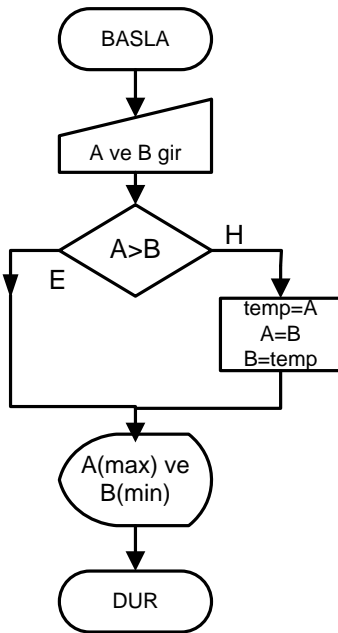
**Örnek :** 1'den 100'e kadar olan sayıların toplamını veren algoritma akış şeması ile oluşturalım.



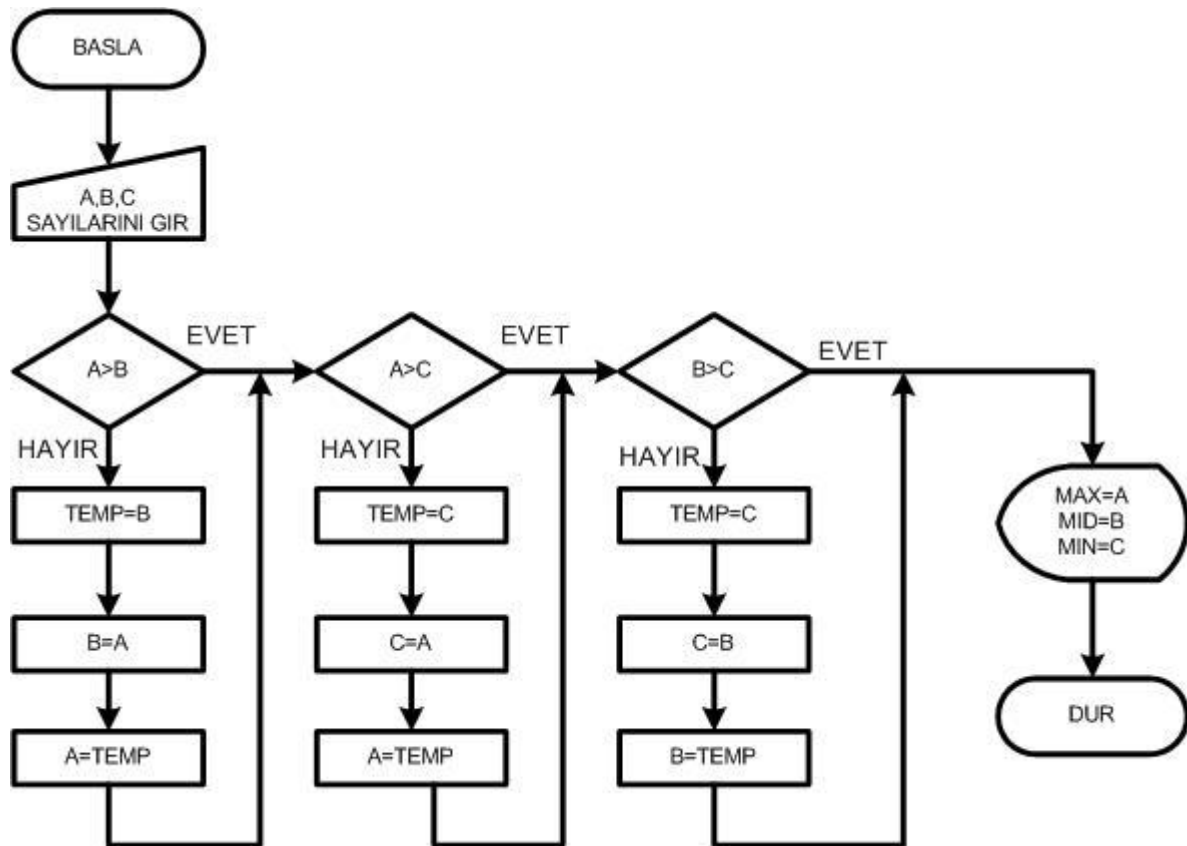
**Örnek:** Klavyeden girilen sayının Negatif /pozitif olduğunu bulan programın akış diyagramını oluşturunuz?



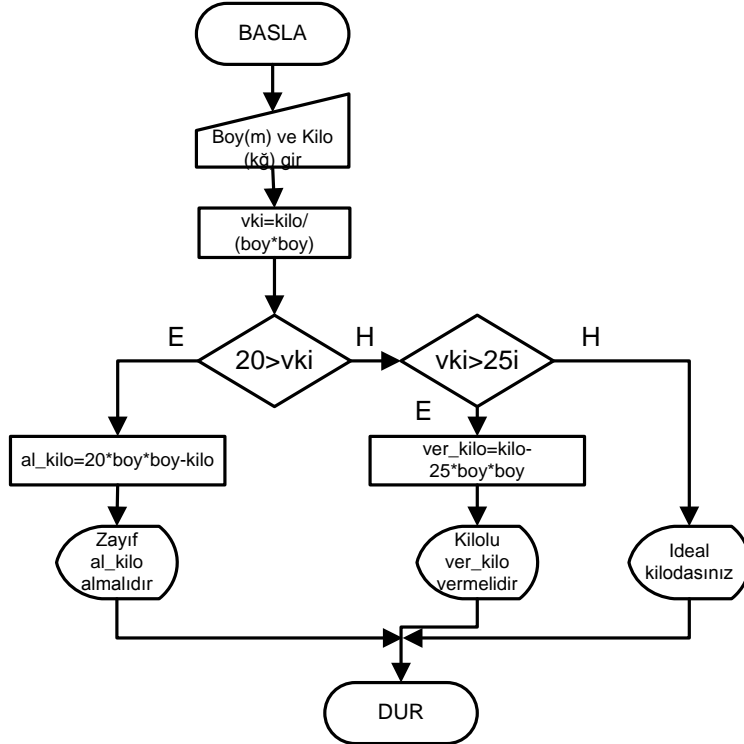
**Örnek:** Rastgele verilen iki sayıyı büyükten küçüğe doğru sıralayıp yazan programın akış diyagramını oluşturunuz



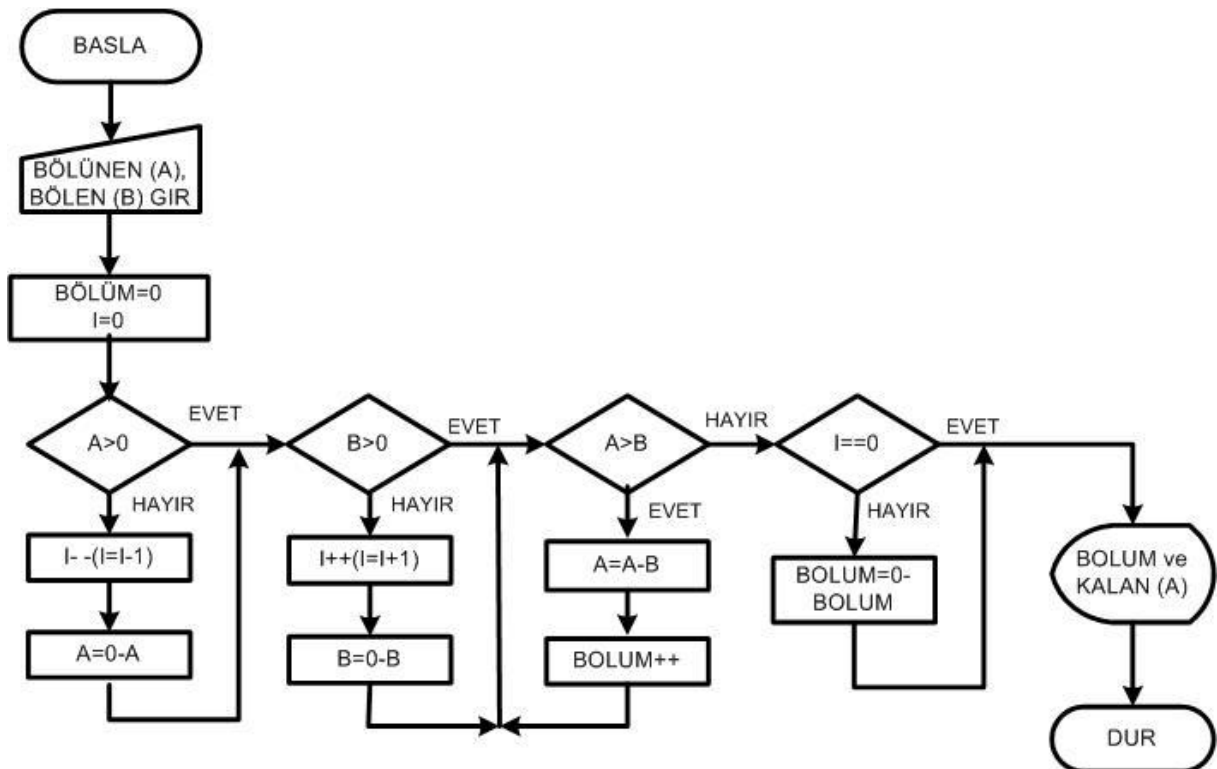
**Örnek:** Rastgele verilen üç sayıyı büyükten küçüğe doğru sıralayıp yazan programın akış diyagramını oluşturunuz?



**ÖRNEK:** Bir kişinin vücut kitle indeksi (vki) şöyle hesaplanmaktadır.  
 $vki = \text{kilo(kg)} / (\text{boy(m)} * \text{boy(m)})$  şayet  $vki < 20$  ise kişi zayıf  $vki > 25$  ise kilolu,  $20 < vki < 25$  aralığında ise ideal kilolu kabul edilmektedir. Buna göre vki hesaplatan programın algoritmasının oluşturunuz. Aynı zamanda kişi zayıf ise ideal kiloya ulaşmak için alması gereken kiloyu, kişi kilolu ise ideal kiloya ulaşmak için vermesi gereken kiloyuda buldurunuz.



**ÖRNEK:** İki Tamsayının bölmesini, bölme işaretini kullanmadan yapan programın akış diyagramını oluşturunuz? (Diyagramda eksiklik var mı? varsa nasıl gideririz?)





**ÖRNEK:** A sayısını B tabanında veren programın algoritmasını akış diyagramı şeklinde oluşturunuz?

A=50 ve B=3 olsun  
Sonuc=1212 olacaktır

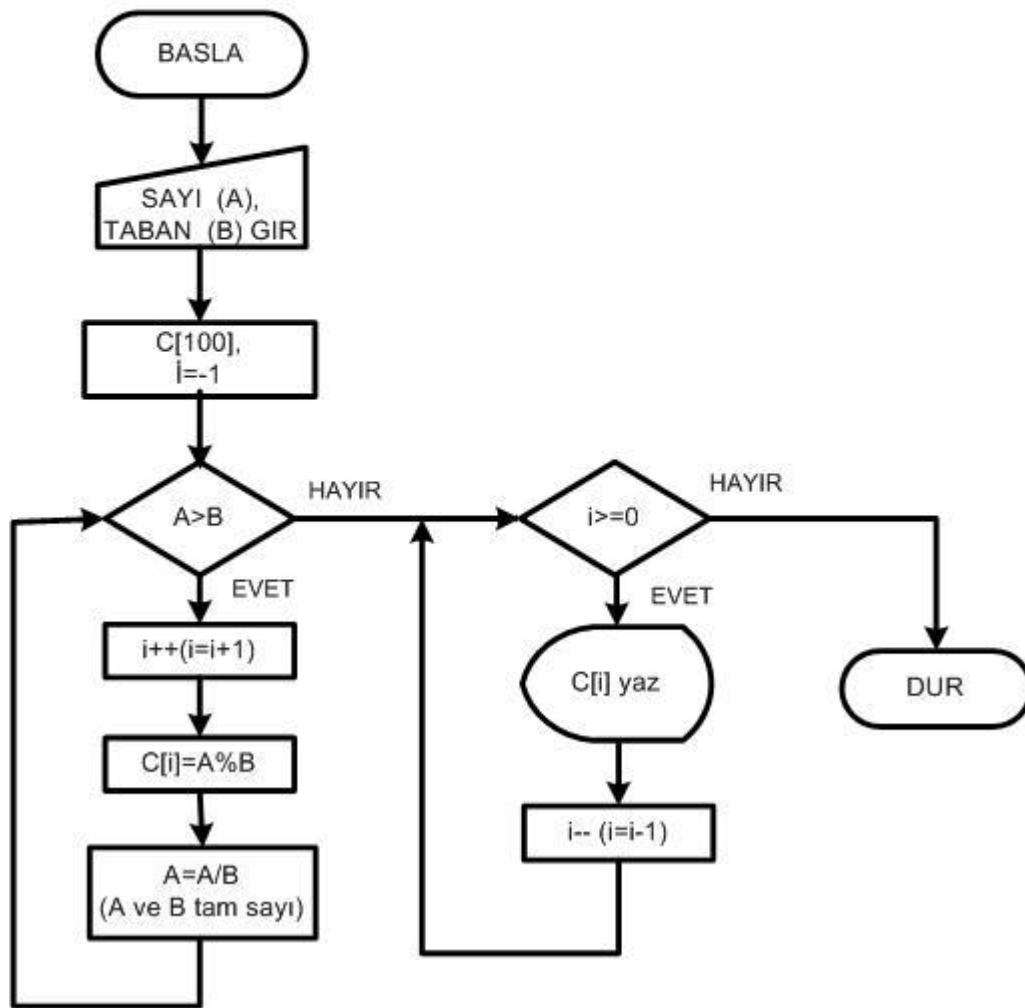
$$\begin{array}{r} 50 \quad | \quad \begin{array}{r} 3 \\ \hline 16 \\ \hline 1 \end{array} \\ \hline 2 \end{array}$$

$$\begin{array}{r} \quad \quad | \quad \begin{array}{r} 3 \\ \hline 5 \\ \hline 2 \end{array} \\ \hline 1 \end{array}$$

$$\begin{array}{r} \quad \quad \quad | \quad \begin{array}{r} 3 \\ \hline 1 \end{array} \\ \hline 2 \end{array}$$

(50)<sub>10</sub>=(1212)<sub>3</sub>

- 3<sup>0</sup>x2=2
- 3<sup>1</sup>x1=3
- 3<sup>2</sup>x2=18
- 3<sup>3</sup>x1=27



#### KAYNAKLAR

- Şevki DEMİRBAŞ, Oğuz ÜSTÜN ve Nihat ÖZTÜRK, Beltek kursu ders notları, Gazi Üniversitesi
- Pappas C.H. ve Murray W.H., “C/C++ Programcının Rehberi”, Sistem Yayıncılık, 2000.
- Pektaş H., “C Dili Kullanarak Bilgisayar Programlama”, KOU EHM, Ders Notları.