



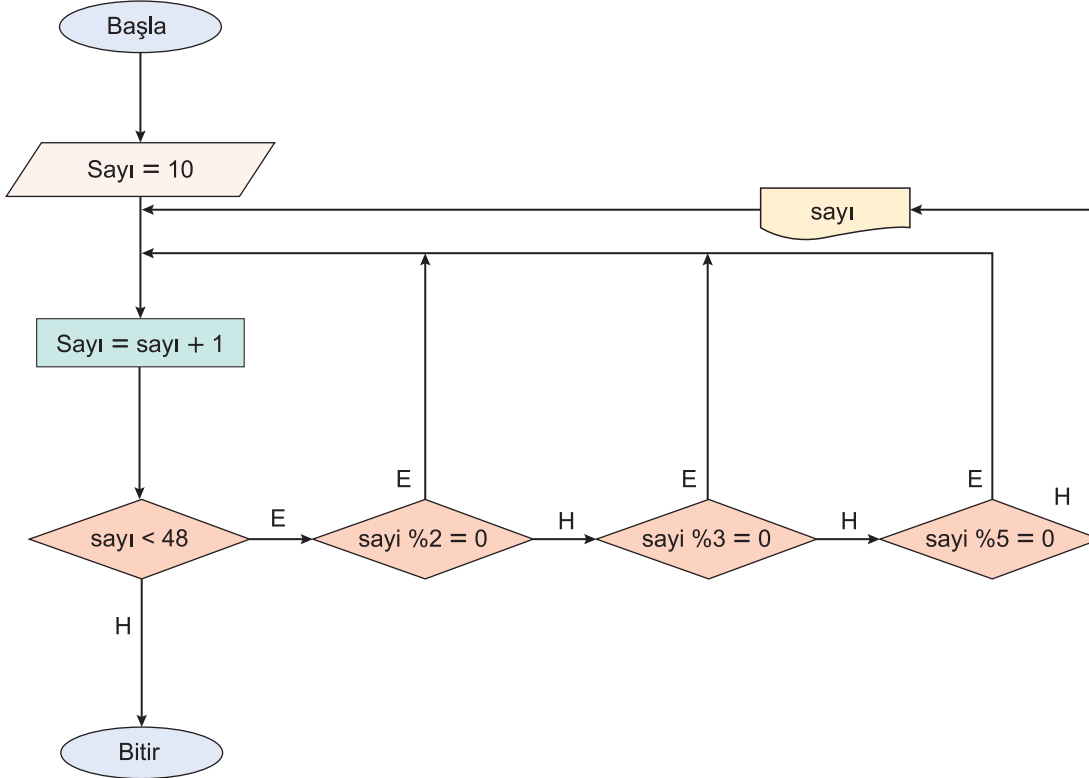
Etkinlik-1

10 ile 48 arasındaki asal sayıları gösteren programın algoritmik doğal dili ve akış şeması aşağıda gösterilmiştir.

Algoritmik Doğal Dil

1. Başla
2. Sayı = 10
3. Sayı = sayı + 1
4. Eğer sayı < 48 ise devam et, değilse 9'a git.
5. Eğer sayı %2 = 0 ise 3. adıma git değilse devam et.
6. Eğer sayı %3 = 0 ise 3. adıma git değilse devam et.
7. Eğer sayı %5 = 0 ise 3. adıma git değilse devam et.
8. Yazdır sayı, 3'e git.
9. Bitir.

Akış Diyagramı





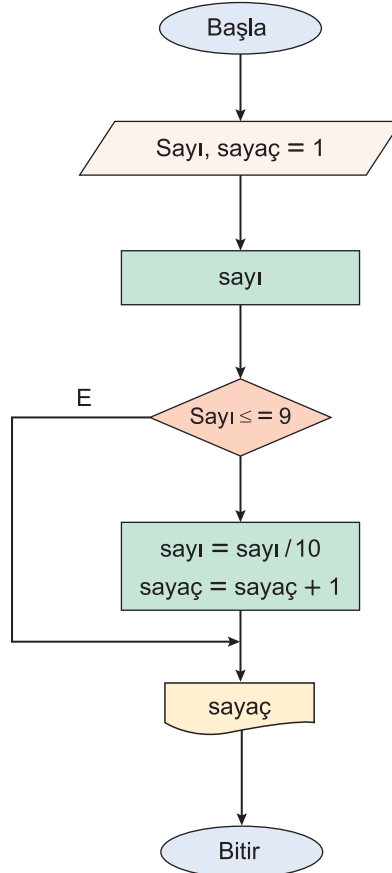
Etkinlik-2

Girilen sayının kaç basamaklı olduğunu söyleyen programın algoritmik doğal dili ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. Sayı, sayaç = 1
3. Sayı gir
4. Eğer $\text{sayı} \leq 9$ ise 6'ya git değilse devam et.
5. $\text{sayı} = \text{sayı} / 10$, $\text{sayaç} = \text{sayaç} + 1$ 4'e git.
6. Yazdır sayaç.
7. Bitir.

Akış Diyagramı



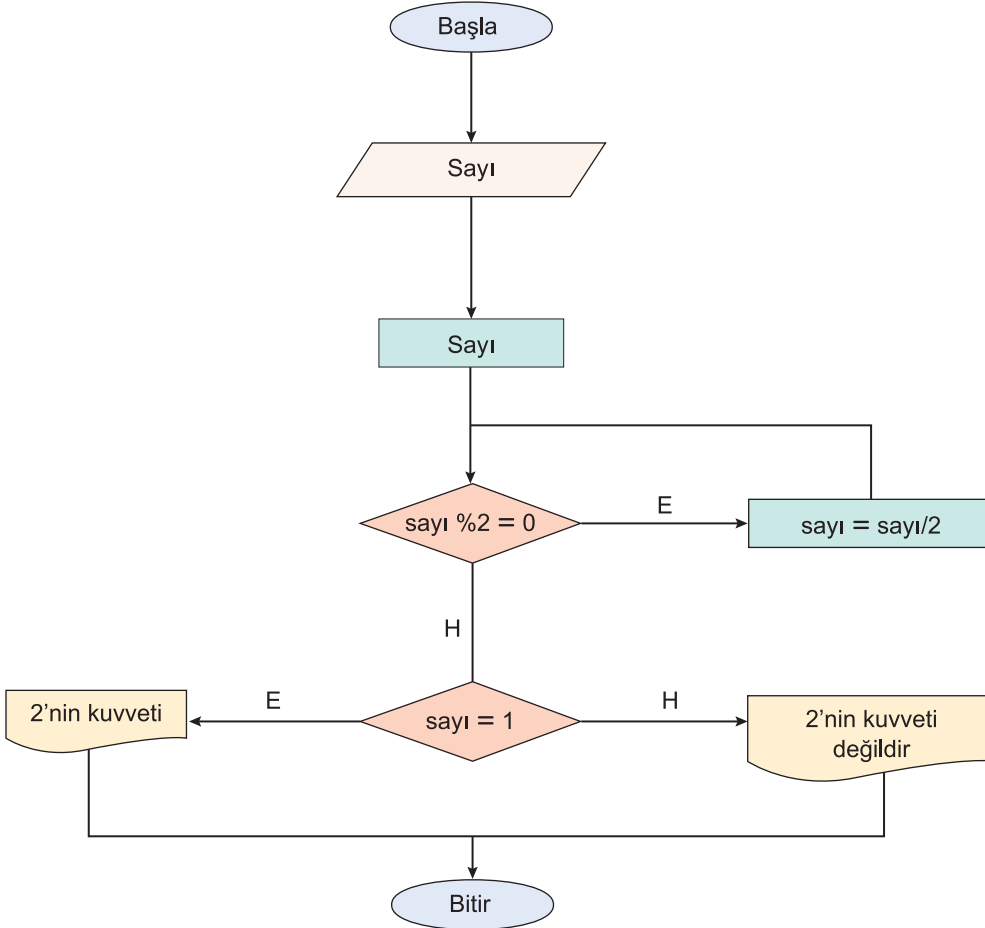
Etkinlik-3

Girilen sayının 2'nin kuvveti olup olmadığını bulan programın algoritma ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. Sayı
3. Sayı gir
4. Eğer $\text{sayı} \% 2 = 0$ ise $\text{sayı} = \text{sayı}/2$ 'e git, değilse devam et.
5. Eğer $\text{sayı} = 1$ ise yazdır "2'nin kuvvetidir." değil ise yazdır "2'nin kuvveti değildir."
6. Bitir.

Akış Diyagramı



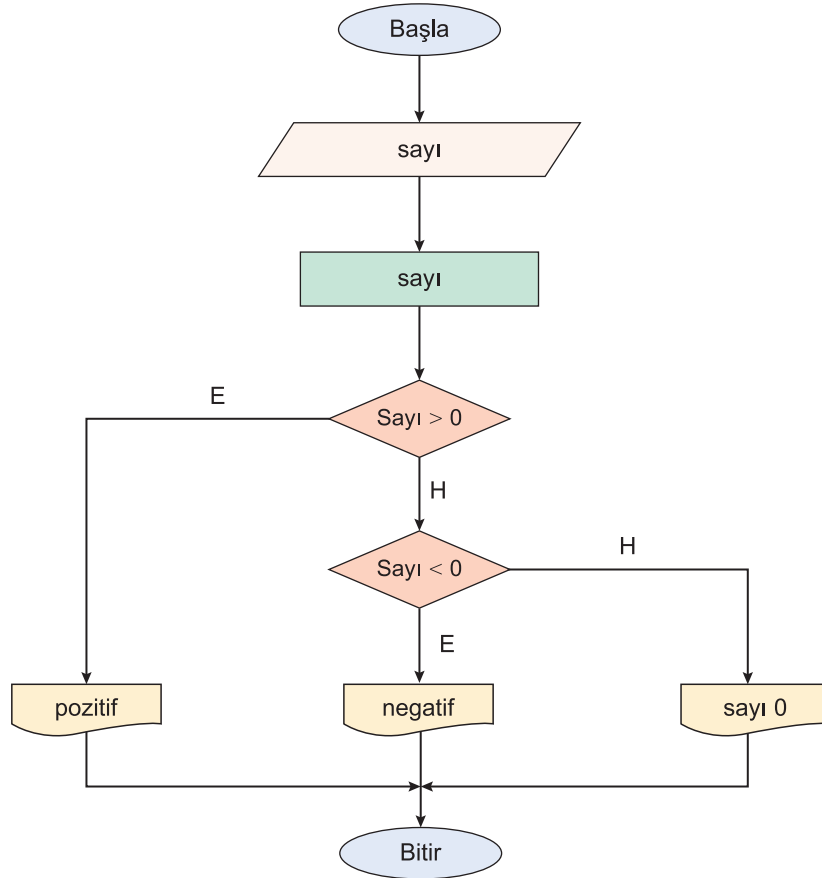
Etkinlik-4

Girilen bir sayının negatif, pozitif veya 0 olup olmadığını bulan programın algoritması ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. Sayı
3. Sayı gir
4. Eğer $\text{sayı} > 0$ ise yazdır "pozitif", değilse devam et.
5. Eğer $\text{sayı} < 0$ ise yazdır "negatif", değilse yazdır "sayı 0".
6. Bitir.

Akış Diyagramı



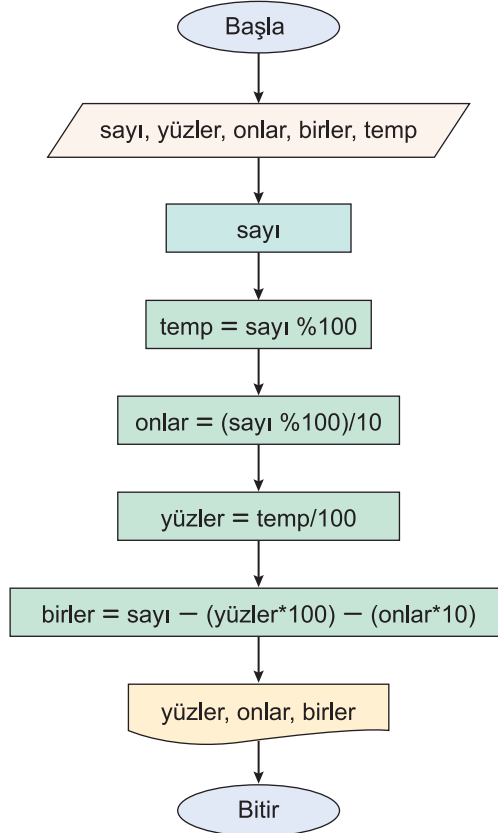
Etkinlik-5

3 basamaklı bir sayının birler, onlar, yüzler basamağındaki sayıları bulan programın algoritması ve akış diyagramı aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. Sayı, yüzler, onlar, birler, temp
3. Sayı gir
4. $temp = sayı \%100$
5. $Onlar = (sayı \%100)/10$
6. $Yüzler = temp/100$
7. $birler = sayı - (yüzler*100) - (onlar*10)$
8. Yazdır yüzler, onlar, birler
9. Bitir.

Akış Diyagramı



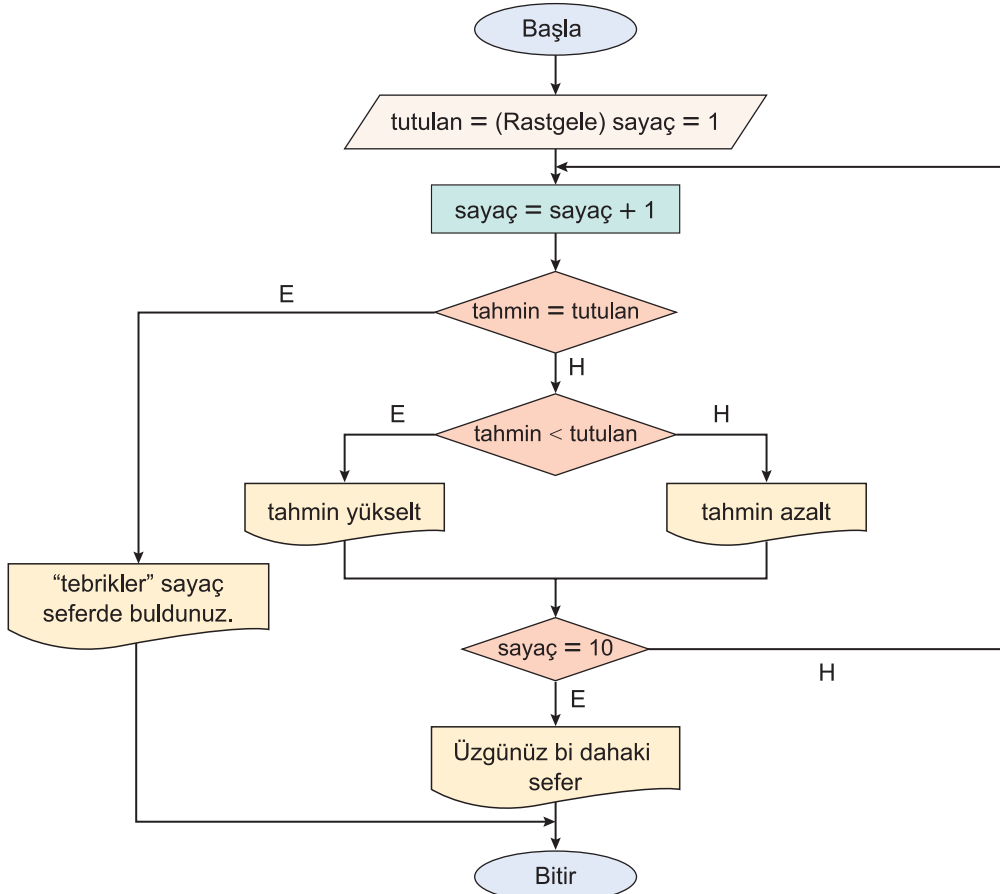
Etkinlik-6

Rastgele tutulan sayıyı 10 seferde bulan programın algoritması ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. tutulan = (rastgele), sayaç = 1
3. Tahmin gir
4. Eğer tahmin = tutulan ise 8'e git değilse devam et
5. Eğer tahmin < tutulan ise yazdır "tahmini yükselt" değilse "tahmini azalt"
6. Eğer sayaç = 10 ise devam et, değilse sayaç = sayaç + 1 3'e git
7. Yazdır "üzgünüz bir dahaki sefere"
8. Yazdır "tebrikler" sayaç "seferde buldunuz"
9. Bitir.

Akış Diyagramı



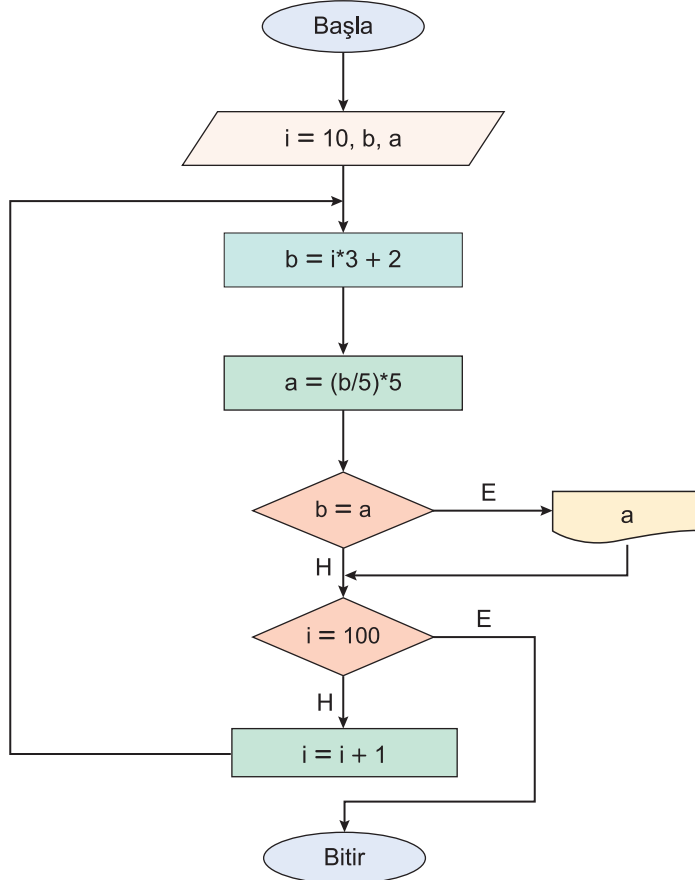
Etkinlik-7

10 ile 100 arasındaki tam sayılardan 3 katının 2 fazlası 5 ile tam bölünebilen sayıları gösteren programın algoritması ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. $i = 10, b, a$
3. $b = i * 3 + 2$
4. $a = (b/5) * 5$
5. Eğer $b = a$ ise yazdır a değilse devam et.
6. Eğer $i = 100$ ise 8'e git değilse devam et.
7. $i = i + 1$ 3. adıma git.
8. Bitir.

Akış Diyagramı

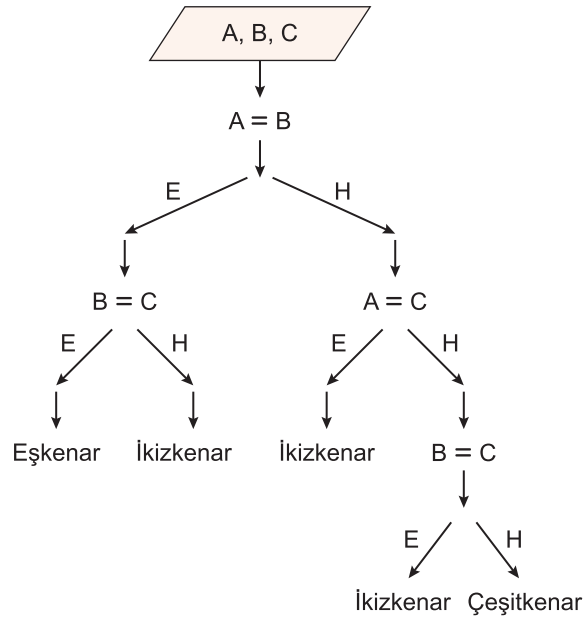




Etkinlik-8

İç açıları verilen üçgenin karar ağacı ve algoritmik doğal dili aşağıda verilmiştir.

Karar Ağacı



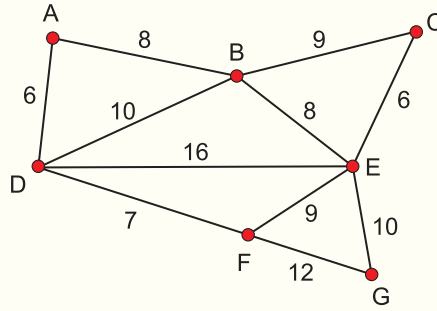
Algoritmik Doğal Dil

1. Başla
2. a, b, c
3. a, b, c gir.
4. Eğer $b = a$ ise 5'e git, değilse 6'ya git.
5. Eğer $b = c$ ise yazdır "Üçgen eşkenardır." 5'e git, değilse yazdır "üçgen ikizkenardır" 8'e git.
6. Eğer $a = c$ ise yazdır "Üçgen ikizkenardır." 8'e git, değilse devam et.
7. Eğer $b = c$ ise yazdır "Üçgen ikizkenardır.", değilse "Üçgen çeşitkenardır."
8. Bitir.

Etkinlik-9

Kraskal algoritması aşağıdaki gibi ifade edilir.

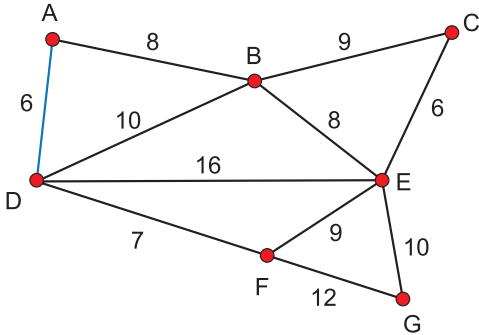
1. Çizgeye ait hatlar maliyetlerine göre sıralanarak bir kuyruk oluşturur.
2. En küçük değerli hattı kuyruktan al ve ağaca ekle.
3. Alınan hat döngü oluşturursa kabul etme.
4. Tüm hatlar için bu işlemleri tekrarla.
5. Eğer kuyruk boşsa sonlandır.



Yukarıdaki çizgeye ait en az maliyetli kapsar ağacının bulunması için aşağıdaki adımlar uygulanır.

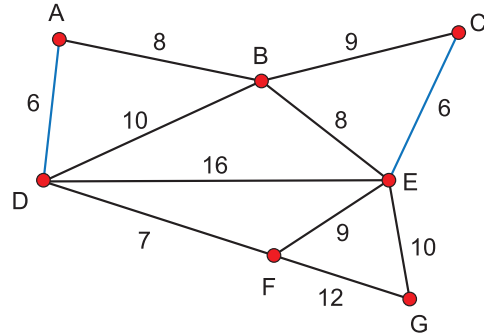
1

En düşük maliyet 6 değeriyle AD ve CE hatlarına ait, AD rastgele olarak seçilir.



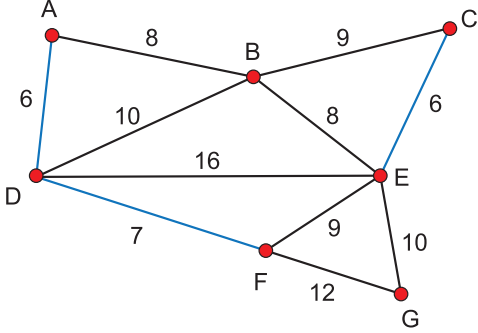
2

Herhangi bir döngü oluşturmadığı için CE hattı da seçilir.



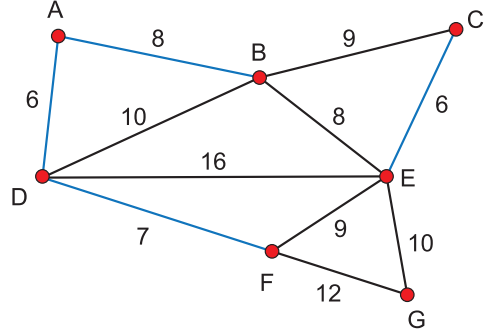
3

Bir sonraki en düşük maliyeti 7 değeri ile DF hattı oluşturuyor.



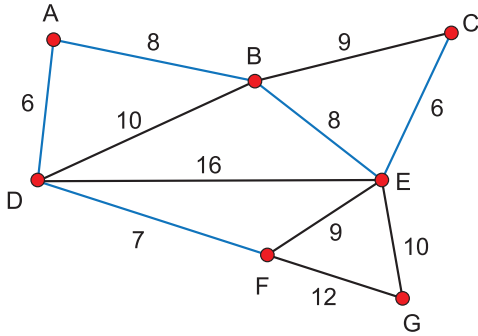
4

AB ve BE hatları 8 değeri ile en düşük maliyete sahip hatlar AB kenarı rastgele seçilerek ağaca ekleniyor.



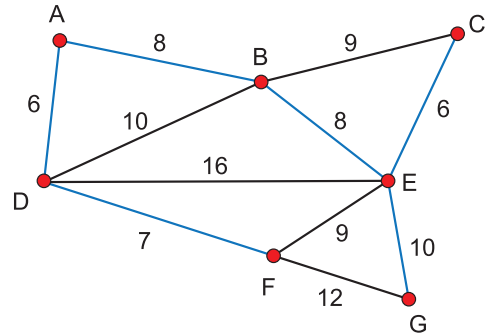
5

Herhangi bir döngü oluşturmadığı için BE hattı da ağaca ekleniyor.



6

Kalan hatlarda BC, EF ve DB seçildiğinde oluşturacağı için bu hatlar seçim dışı bırakılıyor. Dolayısıyla kalan hatlardan daha düşük maliyetle olan EG hattı seçiliyor. Bu işlemin ardından bütün düğümler ağaca dahil olduğu için algoritma tamamlanıyor ve en az maliyetli kapsar ağaç bulunmuş oluyor. 45 değeri ile söz konusu ağaç, çizgedeki tüm düğümleri içeren alternatif ağaçlar arasındaki toplam maliyeti on düşük olandır.



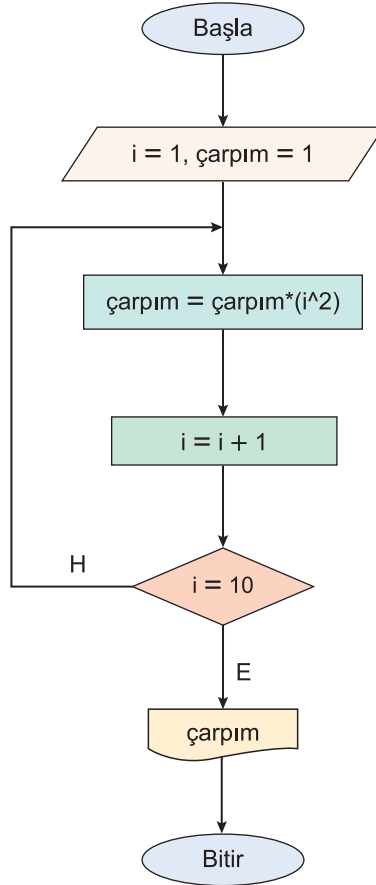
Etkinlik-10

1 ile 10 arasındaki tam sayıların karelerinin çarpımını bulan programın algoritması ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. $i = 1$, $\text{çarpım} = 1$
3. $\text{çarpım} = \text{çarpım} * (i^2)$
4. $i = i + 1$, 3'e git
5. Eğer $i = 10$ ise devam et, değilse 3'e git.
6. Yazdır çarpım
7. Bitir.

Akış Diyagramı





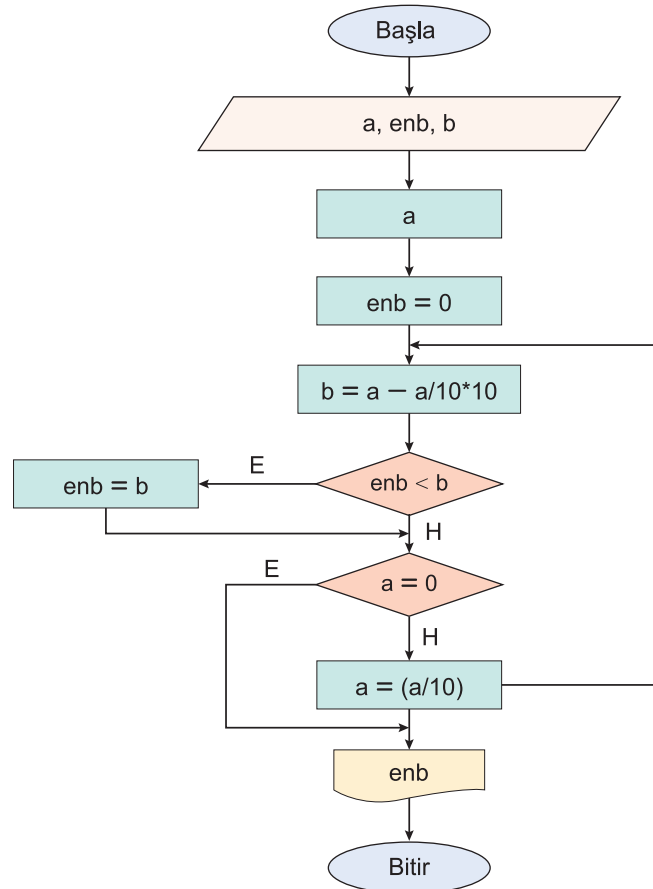
Etkinlik-11

Girilen bir tam sayının basamaklarındaki en büyük sayıyı bulan algoritma ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. Başla
2. a, enb, b
3. a gir
4. $enb = 0$
5. $b = a - a/10*10$
6. Eğer $enb < b$ ise $enb = b$
7. Eğer $a = 0$ ise 9. adıma git
8. $a = (a/10)$, 4. adıma geri dön
9. Yaz enb
10. Bitir

Akış Diyagramı



Etkinlik-12

İki basamaklı ab doğal sayının 5 ile tam bölünüp bölünmediğini bulan algoritmanın işleyişi sözde kodla ve akış şemasıyla gösterilmiştir.

Sözde Kod

Girdi: İki basamaklı ab doğal sayı

Çıktı: ab'nin 5'e tam bölünüp bölünmediğine dair doğru ya da yanlış şeklinde bir çıktı

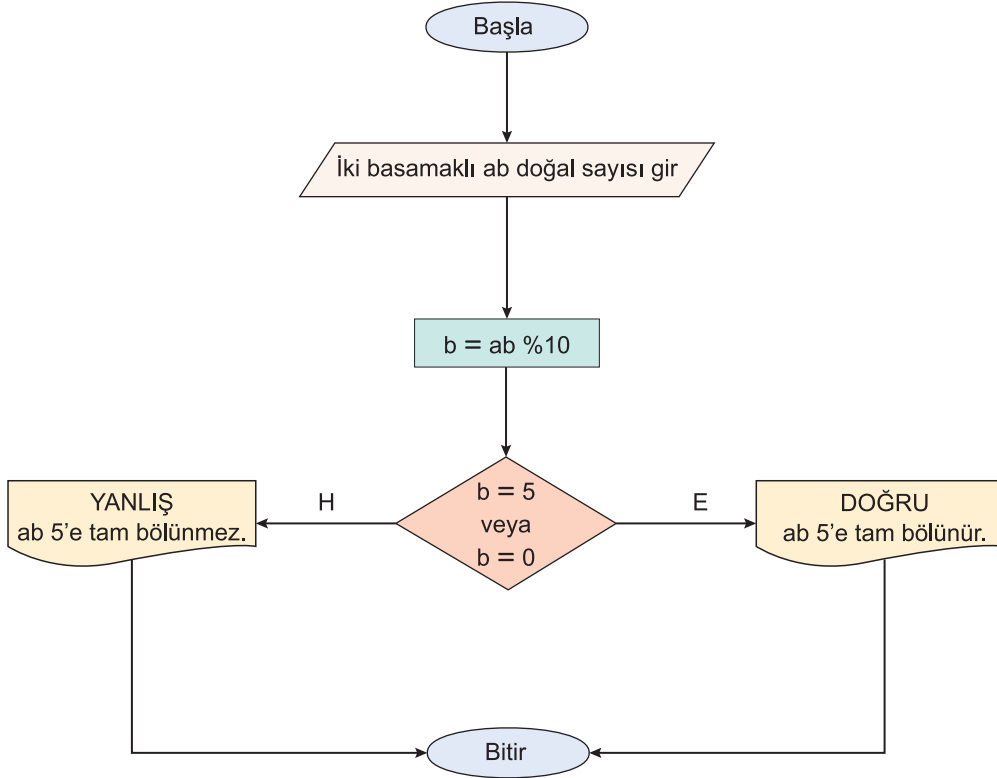
Başla: $b = ab \% 10$ sayının birler basamağını bul

Eğer b 0 veya 5 ise yazdır DOĞRU (ab, 5'e tam bölünür.)

Eğer b 0 veya 5 değilse yazdır YANLIŞ (ab, 5'e tam bölünmez.)

Bitir.

Akış Şeması





Etkinlik-13

a, b, c ve d pozitif tam sayıları için $EBOB(a,b) = x$, $EKOK(c,d) = y$ değerleri bulunup $y^x = t$ sayısı elde ediliyor. t çift ise yeşil ışık, tek ise kırmızı ışık yakan algoritmanın sözde kod ve akış şemasıyla gösterimi aşağıda verilmiştir.

Sözde Kod

Girdi: a, b, c ve d pozitif tam sayısı

Çıktı: t'nin tek ya da çift olması

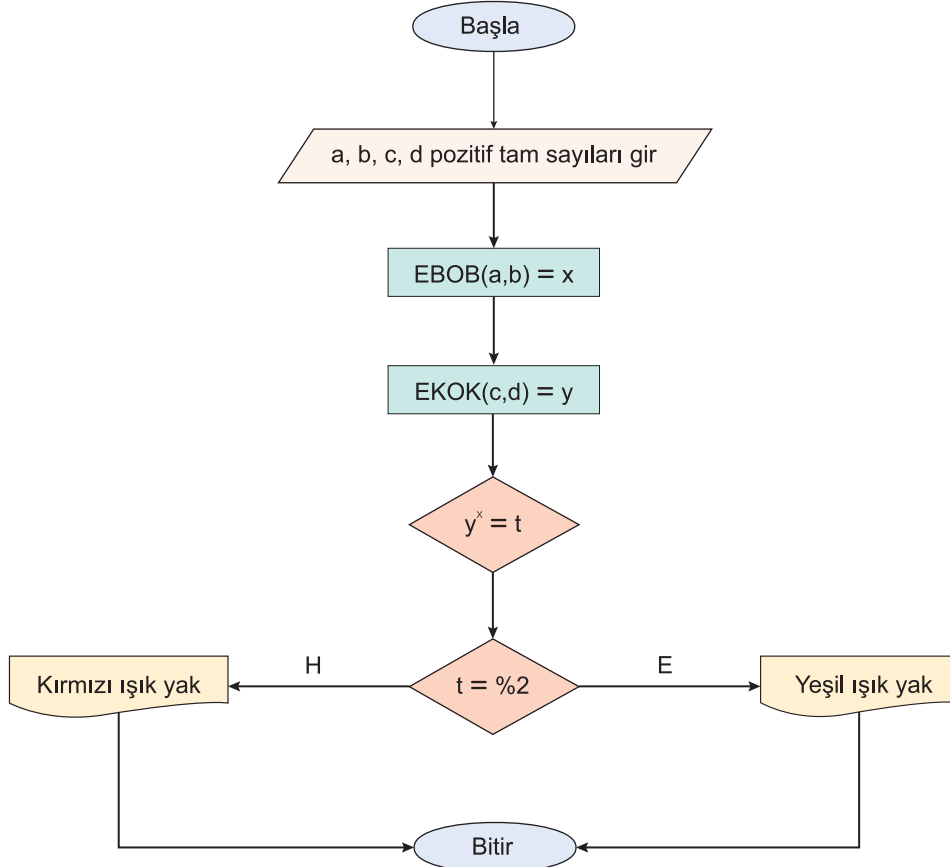
Başla: $EBOB(a,b) = x$ sayısı bul

$EKOK(c,d) = y$ sayısını bul

$y^x = t$ sayısını bul. t çift ise yeşil ışık yanar değilse kırmızı ışık yanar.

Bitir.

Akış Şeması



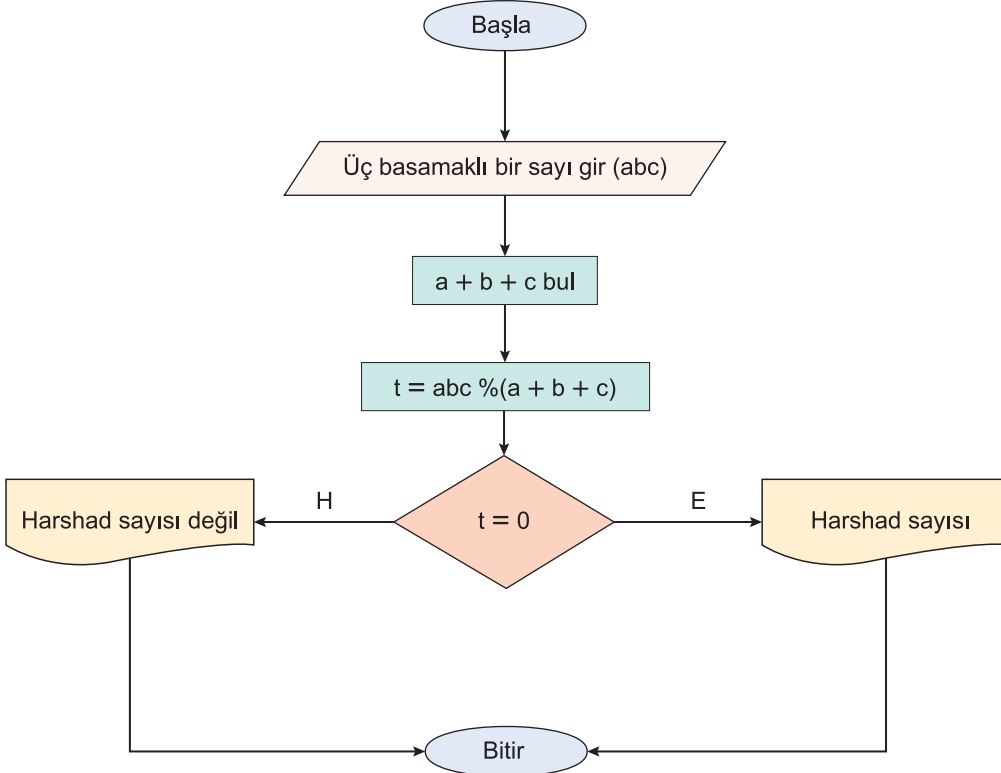
Etkinlik-14

Üç basamaklı bir doğal sayı rakamlarının toplamına tam bölünüyorsa bu sayılara Harshad sayıları denir. Aşağıda bir sayının Harshad sayısı olup olmadığını belirleyen bir algoritmanın işleyişi algoritmik doğal dil ve akış şeması ile gösterilmiştir.

Algoritmik Doğal Dil

1. adım: Başla
2. adım: Girdilerin alınması
Kullanıcıdan üç basamaklı bir sayı girişi yapılması istenir.
3. adım: Rakamlar toplamının hesaplanması. Sayısının rakamları toplanır.
4. adım: Sayının, rakamlarının toplamına bölünmesi.
Sayı rakamlarının toplamına bölünür.
5. adım: Sonucun yazdırılması.
4. adımda elde edilen sonuç ekrana yazdırılır.
6. adım: Bitir.

Akış Şeması





Etkinlik-15

Bölme işlemini çıkarma kullanarak yapan algoritma ve akış şeması aşağıda verilmiştir.

Algoritmik Doğal Dil

1. adım: Başla
2. adım: Bölünen, bölen, sayaç = 0
3. adım: bölünen, bölen gir
4. adım: bölünen = bölünen - bölen
5. adım: Eğer bölünen < bölen ise 6'ya git, değilse sayaç = sayaç + 1, 4'e git.
6. adım: Yazdır sayaç
7. adım: Bitir.

Akış Şeması

