

1. Mantıksal işleçler hangileridir? Giriş ve çıkış değerlerini belirtiniz. (10 PUAN)
2. Aşağıda verilen sayıların işaretli sayı sisteminde ikilik düzende karşılıklarını ve en az kaç byte ile ifade edilebileceklerini yazınız. (12 PUAN)

Handwritten solution for Question 2:

10.10 - 11.00  
11.30 - 12.00  
Soru

127  
-1  
129  
-129  
256

256 128 64 32 16 8 4 2 1

0 0 1 1 1 1 1 1 1  
0 0 0 0 0 0 0 0 1  
1 1 1 1 1 1 1 1 1  
0 1 0 0 0 0 0 0 1  
1 0 1 1 1 1 1 1 1  
1 0 0 0 0 0 0 0 0

2 byte 1 byte

işaret biti  
+:0, -:1 olur.  
kayıp olan  
0 0 0 0 0 0 0 0 0  
2 byte gerekir  
diye TAMAM  
yanıt istiyorum  
Ayrıca

Soru 1: XOR'u zorunlu tutmadım ama gerek de  
kayıp olan 0 0 0 0 0 0 0 0 0  
A B XOR  
0 0 0 doğru  
0 1 1  
1 0 1  
1 1 0

3. Banka hesap numaralarının tutarlı olup olmadığını belirleyen Luhn algoritması şu şekildedir:
- En sağdaki (en düşük anlamlı) basamaktan başlayarak sola doğru her ikinci basamağın iki katını alın.
  - i. adım sonunda oluşan yeni serideki her elemanın basamaklarını kendi aralarında toplayın.
  - ii. adım sonunda oluşan yeni serideki elemanların en sağdaki basamak hariç toplamını bulun.
  - Bulunan toplamı 9 ile çarpıp sonucun son basamağını bulun.
  - Banka hesap numarasının son basamağı ile iv. adımda bulunan sonuç aynı ise, hesap numarasının sahte olmadığına karar verilir.

Yukarıdaki algoritma kullanılarak yapılan bir hesaplama örneği olarak verilmiştir:

Hesap numarası:	7	9	9	2	7	3	9	8	7	1	3
Her ikinci basamağın iki katı:	7	<u>18</u>	9	<u>4</u>	7	<u>6</u>	9	<u>16</u>	7	<u>2</u>	3
Basamaklar toplamı:	7	9	9	4	7	6	9	7	7	2	3
Son basamak hariç elemanların toplamı:	67										
Toplamın 9 katının son basamağı:	3 (çünkü 67*9=603)										
Sonuç:	Hesap numarası sahte değildir										

Bu algoritmaya göre kullanıcının gireceği 11 basamaklı bir hesap numarasının sahte olup olmadığına karar veren bir algoritmanın akış şemasını çizin. Kullanıcı hesap numarasını girecektir ve program sayıyı rakamlarına ayırıp bir diziye yerleştirecektir ve dizinin ilk elemanı hesap numarasının ilk basamağı olacaktır. Ara işlemlerin kullanıcıya gösterilmesine gerek yoktur. (40 PUAN)

```

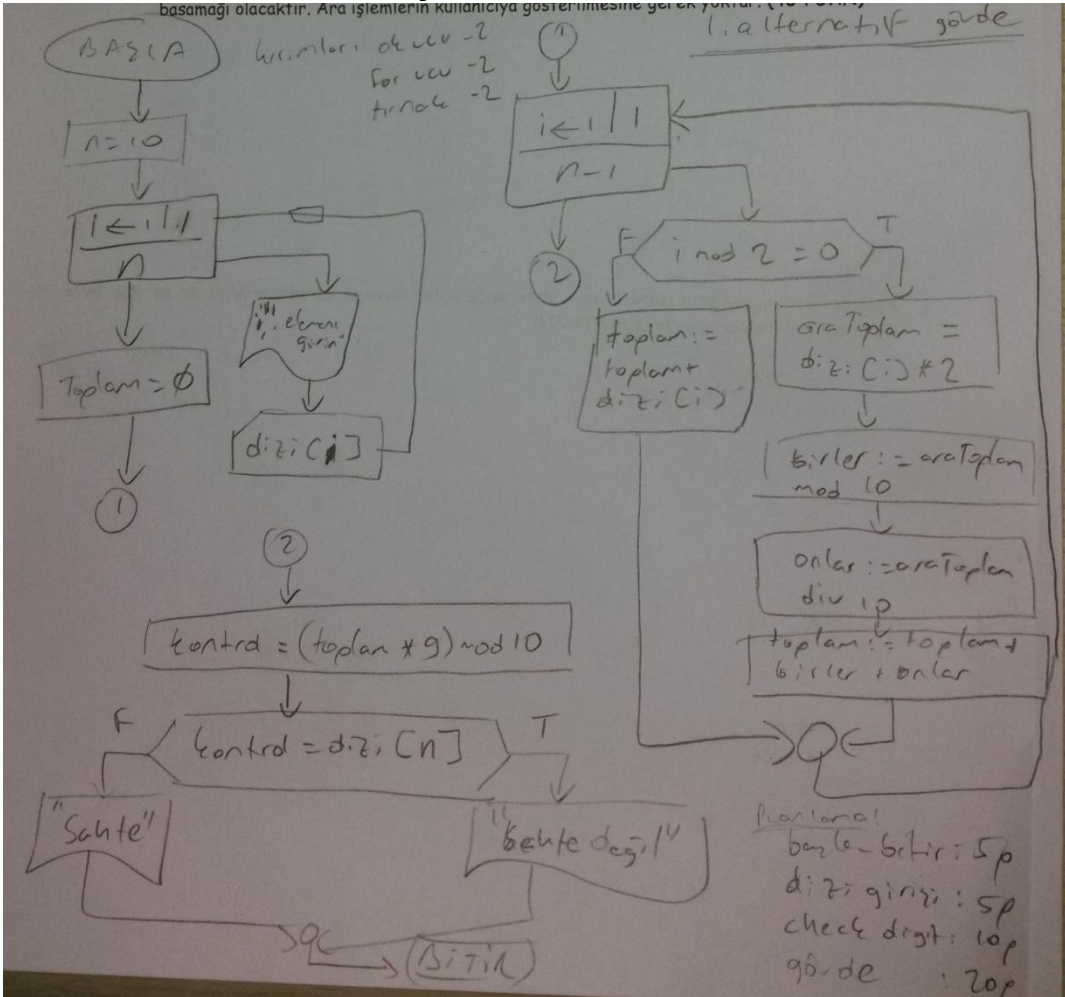
program LuhnV1;
const
    n = 11;
var
    i, toplam, araToplam, birler, onlar, kontrol : integer;

```

```

dizi : Array[1..n] of integer;
begin
  for i := 1 to n do begin
    Write(i, '. elemanı girin: '); ReadLn(dizi[i]);
  end;
  toplam := 0;
  for i:= 1 to n-1 do begin
    if i mod 2 = 0 then begin
      araToplam := dizi[i] * 2;
      birler := araToplam mod 10;
      onlar := (araToplam - birler) div 10;
      toplam := toplam + birler + onlar;
    end
    else begin
      toplam := toplam + dizi[i];
    end;
  end;
  end;
  {WriteLn('Toplam: ', toplam);}
  kontrol := (toplam * 9) mod 10;
  if( kontrol = dizi[n] ) then
    WriteLn('Hesap sahte degildir')
  else
    WriteLn('Hesap sahtedir');
  ReadLn;
end.
{while ile de olur, ara deęerleri dizide tutmak da olur.}

```



N adet elemandan oluşan bir dizinin sıralı olup olmadığını bulan algoritmanın akış diyagramını çizip Pascal dilinde kodunu yazınız.

```

program siraliMi;
const
    uzunluk = 10;
var
    veri : Array[1..uzunluk] of integer;
    i : integer;
    artan, azalan : boolean;
begin
    WriteLn( 'Dizinin sirali olup olmadiginin denetlenmesi.' );
    for i := 1 to uzunluk do begin
        Write( i, '. elemani girin: ' ); ReadLn( veri[i] );
    end;
    artan := true; azalan := true;
    for i := 1 to uzunluk-1 do begin
        if veri[i] > veri [i+1] then    artan := false;
        if veri[i] < veri [i+1] then    azalan := false;
    end;
    if artan then begin
        WriteLn( 'Dizi artan siralidir' );
    end
    else if azalan then begin
        WriteLn( 'Dizi azalan siralidir' );
    end
    else begin
        WriteLn( 'Dizi sirasizdir' );
    end;
    ReadLn( );
end.

program siraliMiV2;
const
    uzunluk = 10;
var
    veri : Array[1..uzunluk] of integer;
    i, sayac : integer;
begin
    WriteLn( 'Dizinin sirali olup olmadiginin denetlenmesi.' );
    for i := 1 to uzunluk do begin
        Write( i, '. elemani girin: ' ); ReadLn( veri[i] );
    end;
    sayac := 0;
    for i := 1 to uzunluk-1 do begin
        if veri[i] < veri [i+1] then
            sayac := sayac + 1;
    end;
    if sayac = uzunluk-1 then begin
        WriteLn( 'Dizi artan siralidir' );
    end
    else if sayac = 0 then begin
        WriteLn( 'Dizi azalan siralidir' );
    end
    else begin
        WriteLn( 'Dizi sirasizdir' );
    end;
    ReadLn( );
end.

```