

## Soru

Çamaşır makinası yıkama süresi belirleme örneğini,

a) Alanların merkezi yöntemini kullanarak çözünüz.

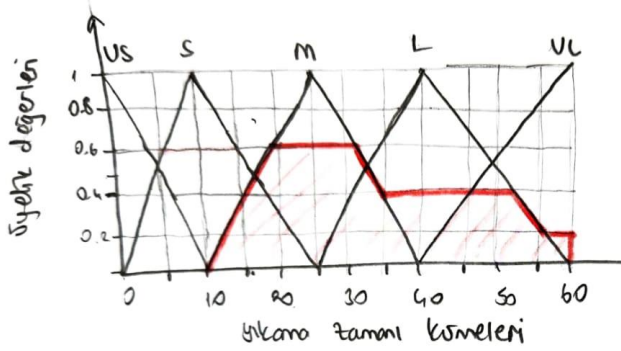
b) Bir MATLAB yazılımı geliştiriniz.

## Cevap

### A şıkkı için çözüm

Problemin giriş ve çıkış kümeleri, kuralları için örnekte verilenler kullanılmıştır.

Durulaştırma aşamasında alanların merkezi yöntemi kullanılarak yapılan soru çözümü aşağıdaki resimde gösterilmiştir.



aktif kuralların uygulanması

$$\begin{aligned} \min\left(\frac{4}{5}, \frac{3}{5}\right) &= \frac{3}{5} \quad N_M \\ \min\left(\frac{4}{5}, \frac{2}{6}\right) &= \frac{2}{5} \quad N_L \\ \min\left(\frac{1}{6}, \frac{2}{6}\right) &= \frac{1}{5} \quad N_{VL} \\ \min\left(\frac{1}{5}, \frac{3}{5}\right) &= \frac{1}{5} \quad N_L \end{aligned}$$

alanların merkezi yöntemi ile durulaştırma:

$$Z = \frac{\frac{3}{5} \times 25 + \frac{2}{5} \times 40 + \frac{1}{5} \times 60 + \frac{1}{5} \times 40}{\frac{3}{5} + \frac{2}{5} + \frac{1}{5} + \frac{1}{5}}$$

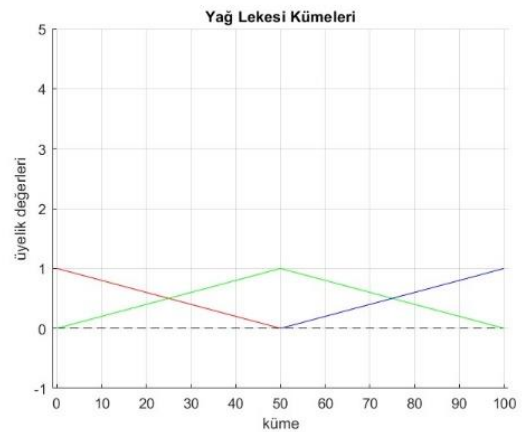
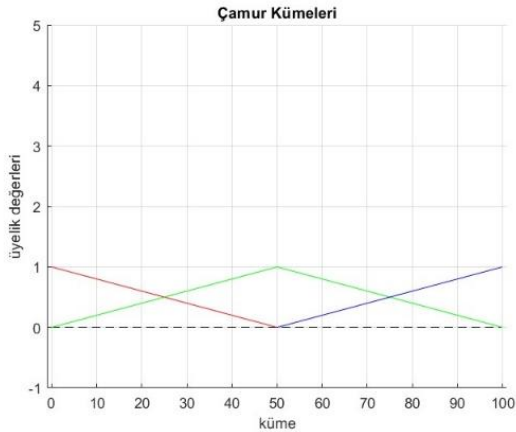
$$Z = \frac{51}{7/5} = \frac{51.5}{7} = 36.4285$$

Yıkama süresi 36.4285 olarak bulunur.

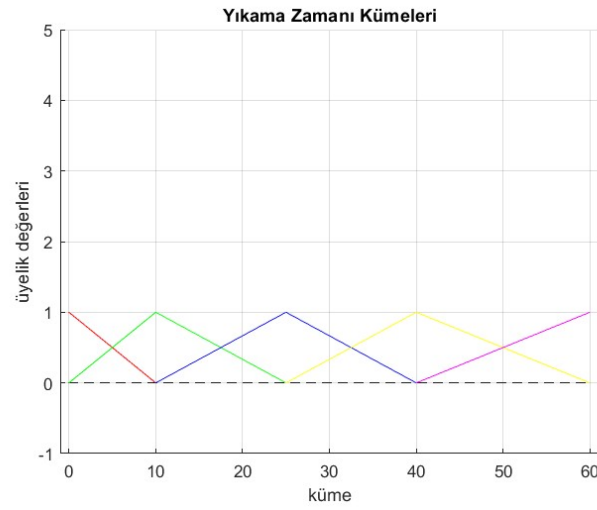
### B şıkkı için çözüm

Çamaşır makinesi yıkama süresi belirleme örneği için MATLAB yazılımı ve çıktıları

Giriş kümeleri,



## Çıkış kümesi



Üyelik fonksiyonu tanımlanması adımı, (Burada calculateMembership fonksiyonu içerisinde üçgen üyelik fonksiyonunun formülü kodlanmıştır.)

```
function membership = calculateMembership(X, trianle)
    x1 = trianle(1);
    xT = trianle(2);
    x2 = trianle(3);

    a=(X-x1)/(xT-x1);
    b=(x2-X)/(x2-xT);
    c=0;
    membership =max(min(a,b),c);
    disp(membership);
    return
end
```

Kural tablosu,

```
% ***** KURAL TANIMLANMASI *****
% kurallar
%      SD    MD    LD
% NG   [  VS    M    L
% MG    S     M    L
% LG    M     L   VL ]

% VS 0 S 10 M 25 L 40 VL 60
% kural çıkışlarının ait oldukları kümelerin tepe değerleri
K = [ 0 25 40
      10 25 40
      25 40 60];
```

Girişlerin bulanıklaştırılması sonucu çıktılar,

```
% ***** GİRİŞLERİN BULANIKLAŞTIRILMASI *****
input_mud = 60;
disp("Çamur kümeleri için üyelik dereceleri, SD-MD-LD");
MuSD = calculateMembership(input_mud,SD);
MuMD = calculateMembership(input_mud,MD);
MuLD = calculateMembership(input_mud,LD);

input_axunge = 70;
disp("Yağ lekesi kümeleri için üyelik dereceleri, NG-MG-LG");
MuNG = calculateMembership(input_axunge,NG);
MuMG = calculateMembership(input_axunge,MG);
MuLG = calculateMembership(input_axunge,LG);
```

```
Çamur kümeleri için üyelik dereceleri, SD-MD-LD
0

0.8000

0.2000

Yağ lekesi kümeleri için üyelik dereceleri, NG-MG-LG
0

0.6000

0.4000
```

Kuralların işlenmesi,

```
% ***** KURAL İŞLEME *****
% Kural tablosundaki kuralların her bir çıkışı için girişlerin minimumu alınır
MU = [min(MuNG, MuSD) min(MuNG, MuMD) min(MuNG, MuLD)
      min(MuMG, MuSD) min(MuMG, MuMD) min(MuMG, MuLD)
      min(MuLG, MuSD) min(MuLG, MuMD) min(MuLG, MuLD) ]

% ***** ÖNERME SONUÇLARI *****
% kural tablosundaki her bir çıkış ile o çıkışın ait olduğu çıkış kümesinin
% tepe değerinin x değeri (K değişkeninde verilmişlerdir) çarpılır
MUK = [MU(1,1)*K(1,1) MU(1,2)*K(1,2) MU(1,3)*K(1,3)
       MU(2,1)*K(2,1) MU(2,2)*K(2,2) MU(2,3)*K(2,3)
       MU(3,1)*K(3,1) MU(3,2)*K(3,2) MU(3,3)*K(3,3) ]
```

```
MU =

0 0 0
0 0.6000 0.2000
0 0.4000 0.2000

MUK =

0 0 0
0 15 8
0 16 12
```

## Durulaştırma adımı,

```
% ***** DURULAŞTIRMA SONUÇLARI *****
% alanların toplamı yöntemi ile durulaştırma
sum_of_areas(MU,MUK);

% maksimum ortalaması
maximum = max(max(MU));
z1 = maximum*15+10;
z2 = -(maximum*15-40);
zMORT = (z1+z2)/2;
disp("maksimum ortalama yöntemi ile asıl çıktı: ");
disp(zMORT);

function sum_of_areas(MU, MUK)
    TOP1 = 0;
    TOP2 = 0;
    for n=1:3
        for m=1:3
            TOP1 = TOP1 + MUK(n,m);
            TOP2 = TOP2 + MU(n,m);
        end
    end
    disp("alanların toplamı yöntemi ile asıl çıktı: ");
    z = TOP1/TOP2
end
```

alanların toplamı yöntemi ile asıl çıktı:

z =

36.4286

maximum ortalama yöntemi ile asıl çıktı:

25

Görüldüğü üzere hem MATLAB kodunda hem de manuel çözümde durulaştırma adımı, alanların toplamı yöntemi ile yıkama süresi 36.429 civarı bir sonuç çıkmıştır. Bu değer maksimum ortalama yöntemi ile çözüldüğünde 25 çıkmıştır.

## Örnek için MATLAB yazılım kodları,

```
% ***** GİRİŞ VE ÇIKIŞ KÜMELERİNİN GÖSTERİMİ *****
% Çamur kümesi
MUD = [0,10,20,30,40,50,60,70,80,90,100];
SD = [0,0,50];
MD = [0,50,100];
LD = [50,100,100];

MUD_ = [SD;MD;LD;];
mud_title = "Çamur Kümeleri";

% yağ lekesi kümesi
AXUNGE = [0,10,20,30,40,50,60,70,80,90,100];
NG = [0,0,50];
MG = [0,50,100];
LG = [50,100,100];

AXUNGE_ = [NG; MG; LG;];
axunge_title = "Yağ Lekesi Kümeleri";

% Yıkama süresi
TIME = [0,10,20,30,40,50,60];
VS = [0,0,10];
S = [0,10,25];
M = [10,25,40];
L = [25,40,60];
VL = [40,60,60];

TIME_ = [VS;S;M;L;VL];
```

```

time_title = "Yıkama Zamanı Kümeleri"

draw_triangles(MUD, MUD_, mud_title);
draw_triangles(AXUNGE, AXUNGE_, axunge_title);
draw_triangles(TIME, TIME_, time_title);

% ***** GİRİŞLERİN BULANIKLAŞTIRILMASI *****
input_mud = 60;
disp("Çamur kümeleri için üyelik dereceleri, SD-MD-LD");
MuSD = calculateMembership(input_mud,SD);
MuMD = calculateMembership(input_mud,MD);
MuLD = calculateMembership(input_mud,LD);

input_axunge = 70;
disp("Yağ lekesi kümeleri için üyelik dereceleri, NG-MG-LG");
MuNG = calculateMembership(input_axunge,NG);
MuMG = calculateMembership(input_axunge,MG);
MuLG = calculateMembership(input_axunge,LG);

% ***** KURAL TANIMLANMASI *****
% kurallar
%      SD      MD      LD
% NG   [  VS    M      L
% MG    S      M      L
% LG    M      L      VL ]

% VS 0 S 10 M 25 L 40 VL 60
% kural çıkışlarının ait oldukları kümelerin tepe değerleri
K = [ 0 25 40
      10 25 40
      25 40 60];

% ***** KURAL İŞLEME *****
% Kural tablosundaki kuralların her bir çıkışı için girişlerin minimumu alınır
MU = [min(MuNG, MuSD) min(MuNG, MuMD) min(MuNG, MuLD)
      min(MuMG, MuSD) min(MuMG, MuMD) min(MuMG, MuLD)
      min(MuLG, MuSD) min(MuLG, MuMD) min(MuLG, MuLD) ]

% ***** ÖNERME SONUÇLARI *****
% kural tablosundaki her bir çıkış ile o çıkışın ait olduğu çıkış kümesinin
% tepe değerinin x değeri (K değişkeninde verilmişlerdir) çarpılır
MUK = [MU(1,1)*K(1,1) MU(1,2)*K(1,2) MU(1,3)*K(1,3)
      MU(2,1)*K(2,1) MU(2,2)*K(2,2) MU(2,3)*K(2,3)
      MU(3,1)*K(3,1) MU(3,2)*K(3,2) MU(3,3)*K(3,3) ]

% ***** DURULAŞTIRMA SONUÇLARI *****
% alanların toplamı yöntemi ile durulaştırma
sum_of_areas(MU,MUK);

% maksimum ortalaması
maximum = max(max(MU));
z1 = maximum*15+10;
z2 = -(maximum*15-40);
zMORT = (z1+z2)/2;
disp("maksimum ortalama yöntemi ile asıl çıktı: ");
disp(zMORT);

function sum_of_areas(MU, MUK)

```

```

TOP1 = 0;
TOP2 = 0;
for n=1:3
    for m=1:3
        TOP1 = TOP1 + MUK(n,m);
        TOP2 = TOP2 + MU(n,m);
    end
end
disp("alanların toplamı yöntemi ile asıl çıktı: ");
z = TOP1/TOP2
end

function draw_triangles(X, U, title_)
    % X: Evren
    % U : kümelerin listesi
    figure;
    hold on;
    plot(X, zeros(size(X)), 'k--'); % X eksenini
    colors = {'r', 'g', 'b', 'y', 'm'};

    for i = 1:length(U)
        x = zeros(1,3);
        y = zeros(1,3); % Y vektörünü sıfırla
        for j = 1:3
            x(j) = U(i,j);
        end
        if i == 1 % sağ sınır üçgeni
            y = [1,1,0];
        elseif i == length(U)
            y = [0,1,1]; % sol sınır üçgeni
        else
            y = [0,1,0]; % ara üçgenler,
        end

        plot(x, y, colors{i}); % üçgen i
    end
    xlabel('küme');
    ylabel('üyelik değerleri');
    title(title_);

    axis([min(X)-1 max(X)+1 -1 5]);
    grid on;
    hold off;
end

function membership = calculateMembership(X, trianle)
    x1 = trianle(1);
    xT = trianle(2);
    x2 = trianle(3);

    a=(X-x1)/(xT-x1);
    b=(x2-X)/(x2-xT);
    c=0;
    membership =max(min(a,b),c);
    disp(membership);
    return
end

```