



**SAKARYA**  
ÜNİVERSİTESİ

ÜNİVERSİTESİ  
SAKARYA

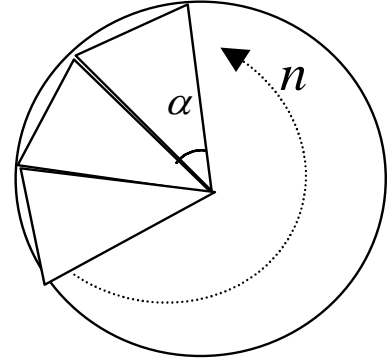
## ***SAYISAL ANALİZ DÖNEM ÖDEVİ***

**B181200559 ARSENE ADJEVİ**

Konu:

Bir matematikçi yandaki şeklin alanını analitik olarak  $A_D =$  bulmuş ve bu şekle daire adını vermiştir.

Sadece ikiz kenar üçgenlerin alanını hesaplayabilen bir bilgisayarla bu dairenin alanını  $\% \epsilon_a < 0.01$  hata ile bulabilecek bir algoritma oluşturun.



Başlangıçta daire dört tane böldüğümüz zaman dört tane ikiz kenar üçgen elde ederiz. Bu dört tane ikiz kenar üçgenlerden bir tane alıp onun alan için genel bir formula çıkartalım.

$OAB$  alalım ve  $A$  onun alanı olsun:

$$A = \frac{AB * OC}{2}$$

$AB$  ve  $OC$  değerleri bilmiyoruz. Önce bu değerleri arayalım.

$$\sin \gamma = \frac{AC}{OA} \Rightarrow AC = OA * \sin \gamma$$

$AB = 2 * AC$  oldğu biliyoruz o zaman

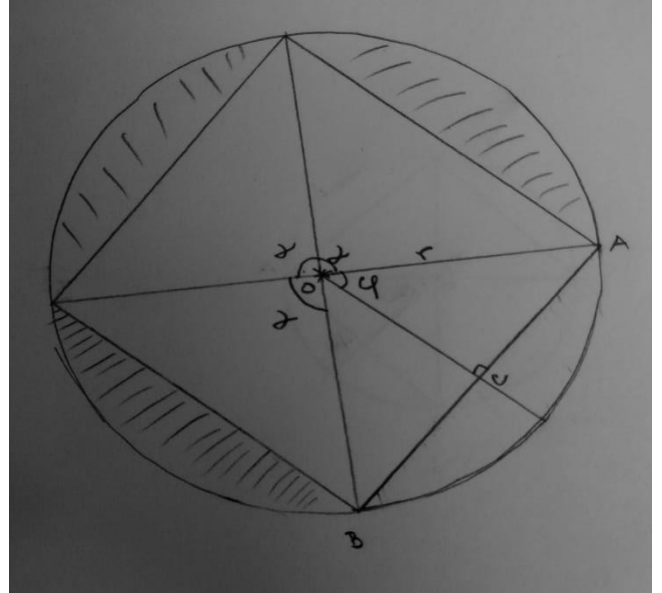
$$AB = 2 * OA * \sin \gamma, OA = r(\text{dairenin yarıçapı})$$

$$AB = 2 * r * \sin \gamma$$

Daire dört paçaya böldük ama yerine genel bir formül bulmak için  $n$  kullanalım

$$\gamma = \frac{\alpha}{2}, \alpha = \frac{2\pi}{n}$$

$$\text{Böylece } \gamma = \frac{\pi}{n}$$



$$AB = 2 * r * \sin \frac{\pi}{n}$$

OC değeri ise:

$$\cos \gamma = \frac{OC}{OA} \implies OC = OA * \cos \gamma$$

$$OC = r * \cos \frac{\pi}{n}$$

Sonunda

$$\text{OAB ikiz üçgenin alanı } A = \frac{2 * r^2 * \sin \frac{\pi}{n} * \cos \frac{\pi}{n}}{2} \rightarrow$$

$$A = r^2 * \sin \frac{\pi}{n} * \cos \frac{\pi}{n}$$

Daire  $n$  tane parçaya böldürse ikiz kenar üçgenlerin alanların toplamı

$$A[n] = n * \left( r^2 * \sin \frac{\pi}{n} * \cos \frac{\pi}{n} \right) \text{ Olur.}$$

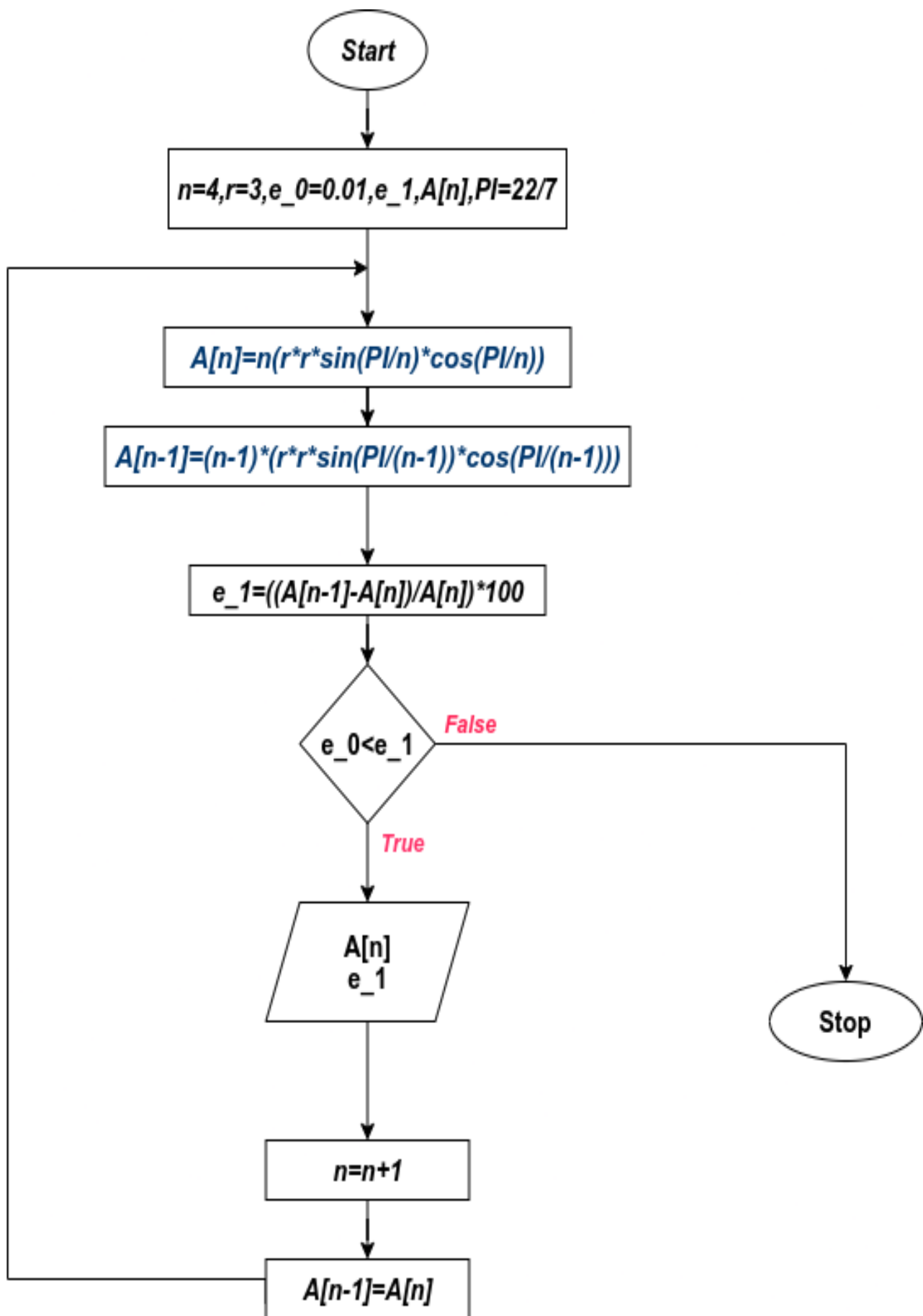
Yazacağımız algoritmaya göre üçgenlerin sayısı artırdıkça elde edeceğimiz hata düşük olur.

Daha önce söyledim gibi  $n$  üçgenlerin sayısı,  $r=3$  dairenin yarıçapı,  $\pi = 22/7$ ,  $e_{-1}$  ilk

hata,  $e_{-0} = 0.01$  olsun. Başlangıçta  $n = 4$  alırsak  $A[4] = 4 * \left( r^2 * \sin \frac{\pi}{4} * \cos \frac{\pi}{4} \right)$  olacak .

ilk hata  $e_{-1} = \left( \frac{A[n]-A[n-1]}{A[n]} \right) * 100$  formül ile bulacağız yani  $n = 4$  olduğunda  $e_{-1} =$

$\left( \frac{A[4]-A[3]}{A[4]} \right) * 100$  olacaktır.  $e_{-1} < e_{-0}$  olana kadar  $n$  bire attıracağız Ve bu şekilde aşağıdaki akış diyagramı çizdiğimiz olur.



Aşağıdaki C programlama dilinde yazdığım kodun ekran alıntısı bulunmaktadır.

```
1 #include<stdint.h>
2 #include<stdio.h>
3 #include<math.h>
4 #define PI 3.142857142857143
5 #ifndef MAX
6 #define MAX 200
7 //Maximal Number of triangle if Max number was not defined on compiling time
8 #endif
9 int main(int argc, char const *argv[])
10 {
11     int n=4,r=3;
12     float e_0=0.01,e_1,A[MAX];
13     /*
14     General Formule after calcule and at the begining a took n=4 for number
15     of the triangle inside the circle.After that we will increase the number
16     for triangle until e_0>e_1.
17     */
18     do
19     {
20         n=n+1;
21         A[n]=n*(pow(r,2)*sin(PI/n)*cos(PI/n));
22         A[n-1] = (n-1)*(pow(r,2)*sin(PI/(n-1))*cos(PI/(n-1)));
23         e_1=fabs((A[n]-A[n-1])/A[n])*100;
24         printf("%d_ Air value:%lf with error: %lf\n",n,A[n],e_1);
25         A[n-1]=A[n];
26     } while (e_0<e_1); //Control if(fabs((A[n]-A[n-1])/A[n])<e_0)
27
28
29     return 0;
30 }
31
```

Kod dosyasındaki */bin/* dosyadaki bulunan **binary code** komut olarak `./donem_odev.out` çalıştırınızda ya kendiniz **gcc** yanda **clang** compiler ile

***gcc -g -Wall donem\_odev.c -lm -o donem\_odev.out*** yada ***clang -g -Wall donem\_odev.c -lm -o donem\_odev.out*** compile yapıp çalıştırabilirsiniz.

Ben kodu çalıştırdığım zaman şu sonuçlar elde ettim:

5\_ Air value:21.402287 with error: 15.896853  
6\_ Air value:23.388374 with error: 8.491774  
7\_ Air value:24.634787 with error: 5.059562  
8\_ Air value:25.463890 with error: 3.255997  
9\_ Air value:26.041615 with error: 2.218467  
10\_ Air value:26.459543 with error: 1.579501  
11\_ Air value:26.771294 with error: 1.164495  
12\_ Air value:27.009855 with error: 0.883239  
13\_ Air value:27.196383 with error: 0.685853  
14\_ Air value:27.344929 with error: 0.543231  
15\_ Air value:27.465120 with error: 0.437615  
16\_ Air value:27.563721 with error: 0.357718  
17\_ Air value:27.645599 with error: 0.296172  
18\_ Air value:27.714325 with error: 0.247979  
19\_ Air value:27.772568 with error: 0.209713  
20\_ Air value:27.822353 with error: 0.178941  
21\_ Air value:27.865238 with error: 0.153901  
22\_ Air value:27.902443 with error: 0.133339  
23\_ Air value:27.934923 with error: 0.116271  
24\_ Air value:27.963449 with error: 0.102013  
25\_ Air value:27.988634 with error: 0.089982  
26\_ Air value:28.010983 with error: 0.079784  
27\_ Air value:28.030901 with error: 0.071059  
28\_ Air value:28.048733 with error: 0.063574  
29\_ Air value:28.064756 with error: 0.057095  
30\_ Air value:28.079210 with error: 0.051475

31\_ Air value:28.092291 with error: 0.046563  
32\_ Air value:28.104168 with error: 0.042261  
33\_ Air value:28.114985 with error: 0.038473  
34\_ Air value:28.124863 with error: 0.035123  
35\_ Air value:28.133909 with error: 0.032155  
36\_ Air value:28.142212 with error: 0.029503  
37\_ Air value:28.149853 with error: 0.027143  
38\_ Air value:28.156900 with error: 0.025030  
39\_ Air value:28.163412 with error: 0.023121  
40\_ Air value:28.169443 with error: 0.021410  
41\_ Air value:28.175039 with error: 0.019862  
42\_ Air value:28.180241 with error: 0.018457  
43\_ Air value:28.185085 with error: 0.017189  
44\_ Air value:28.189602 with error: 0.016022  
45\_ Air value:28.193823 with error: 0.014971  
46\_ Air value:28.197771 with error: 0.014002  
47\_ Air value:28.201469 with error: 0.013114  
48\_ Air value:28.204941 with error: 0.012308  
49\_ Air value:28.208200 with error: 0.011556  
50\_ Air value:28.211267 with error: 0.010872  
51\_ Air value:28.214157 with error: 0.010242  
52\_ Air value:28.216881 with error: 0.009653

**Yanı  $r = 3$  yarıçaplı daire içinde 52 tane ikiz kenarı üçgenler böldürse bu üçgenlerin alanları toplanırsa Dairenin alanı  $\varepsilon = 0.009653$  Hata ile  $AD = 28.216881br^2$  Bulmuş olur.**