

ANKARA UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT
BİLGİSAYAR PROGRAMLAMA II
BAHAR 2023-24

LAB 3-2 Quiz
Prof. Dr. Semra GÜNDÜÇ

Tarih: 08/03/2024

Maclaurin Serisi formunda $\cos(x)$ fonksiyonu aşağıdaki seri formülü ile ifade edilmektedir.

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

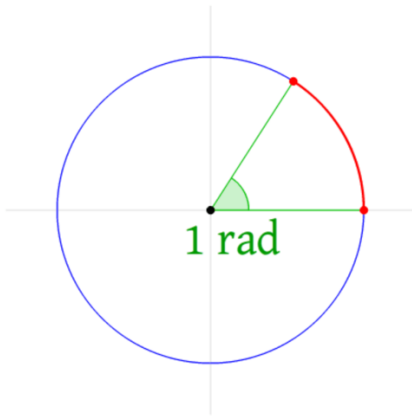
Bu seri

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$$

şeklinde olup sonsuza kadar yapılacak bir toplam şeklindedir.

Kullanıcıdan alınacak bir x değeri ve bu serinin kaç iterasyon sonra duracağını bulmada kullanılacak hassasiyet değeri ile $\cos(x)$ fonksiyonunun değerini hesaplayıp ekrana basan programı yazınız. **x değeri girdi dosyalarında derece olarak size verilmiş olup radyana dönüştürülmek zorundadır.** Programınız ayrıca `<math.h>` kütüphanesinden `cos(x)` fonksiyonunun sonucunu da ekrana yazdırmalıdır. Böylece her iki sonucun yakınsama durumundan sonucunuzun doğruluğunu da kontrol edebilirsiniz.

Radyan, bir dairede yarıçap uzunluğundaki yay parçasını gören merkez açıya eşit açı ölçme birimidir. 1 radyan $180/\pi$ ya da yaklaşık 57,2958 derecedir ($57^\circ 17' 45''$). **Dereceden radyana dönüştürmek için verilen derece değeri $\pi/180$ ile çarpılır. π , `<math.h>` kütüphanesinden `M_PI` sabiti ile ifade edilir.**



Elbette sonsuza kadar bir toplama işlemi yapılamayacağından, bu seri hesabından elde edilecek değer kaç iterasyon yapıldığına bağlı olarak değişecektir. Bu noktada kullanıcıya verilecek bir hassasiyet değeri üzerinden seri hesaplanmasında kaç iterasyon gidileceği programınızda elde edilecektir. Diğer bir deyişle, her bir terimde elde edilecek değer her iterasyonda daha da

azalacağından yüksek iterasyonlarda toplama ilave edilecek değer oldukça küçülecektir. Underflow durumuna düşmemek için ard arda herhangi iki iterasyon arasındaki $\cos(x)$ değerindeki farkın mutlak değeri, girdi dosyasında verilen hassasiyet değerine eşit ya da daha küçük çıktığında hesaplama duracaktır.

Örnek:

Hata payı 0.02 olarak belirlenmiş ve

n. iterasyonda $\cos(x)$ 0.2456

(n+1). iterasyonda $\cos(x)$ 0.2568 ise hesap durdurulabilir.

Hata payı 0.02 olarak belirlenmiş ve

n. iterasyonda $\cos(x)$ 0.2356

(n+1). iterasyonda $\cos(x)$ 0.2668 ise hesap devam etmelidir.

Örnek 1:

Girdi Dosyası:

30 → Radyana Çevirilecek Derece

0.02 → Hassasiyet Değeri

Çıktı Dosyası:

$\cos(30) = 0.866054$

$\cos(30)$ from math.h = 0.866025

Not: Virgülden sonra 6 hane yazdırılmalıdır. Mutlak değer için `<math.h>` kütüphanesinden **fabs()** fonksiyonu kullanılabilir. Kuvvet değeri için **pow()** fonksiyonu kullanılabilir. **Faktöriyel** için hazır kütüphane fonksiyonu kullanmanız yasaktır.

Dosya Teslimi:

1- Dosyanızı öğrenci_numaranız.c olarak isimlendiriniz → örnek 87456933.c

2- Dosyanızı ekampüs sisteminde ilgili yere yükleyiniz.

Derleme İşlemi:

//normal derleme

gcc öğrenci_numaranız.c -o çalıştırılabilirDosyaİsminiz

./çalıştırılabilirDosyaİsminiz

//.txt dosyasının input olarak kullanımı

./çalıştırılabilirDosyaİsminiz<input.txt

// çıktının .txt dosyasına yazdırılması

./çalıştırılabilirDosyaİsminiz<input.txt>myoutput.txt

// iki dosyanın karşılaştırılması

diff -w dosya1 dosya2

Örnek:

gcc 87456933.c -o myprogram

./myprogram<input1.txt>myoutput1.txt

diff -w output1.txt myoutput1.txt

LÜTFEN, SİZE VERİLEN I/O FORMATLARINA DİKKAT EDİNİZ!