Özyinelemeli (recursive) Fonksiyonlar



Kendilerini doğrudan veya dolaylı yoldan çağıran fonksiyonlara öz yinelemeli fonksiyonlar denilir. Döngü işlemleri kullanılarak yapılan iterative işlemler, özyinelemeli fonksiyonlar kullanılarak ta gerçekleştirilebilir. Özyinelemeli fonksiyonlar problemleri daha küçük parçalara indirgeyerek çözme yaklaşımı sunmaktadır.

Örnek: bir tamsayının faktöriyelinin hesaplanması.



.



|  |  |
| --- | --- |
| N! = N\*N-1\*………\*2\*1 | N!=N\*N-1! |
|  | N-1!=N-1\*N-2! |
|  | . |
|  | . |
|  | . |
|  | 2!=2\*1! |
|  | 1!=1 |



int faktoriyel(int n)

{

int faktor;



if (n <= 1)



faktor = 1;

else



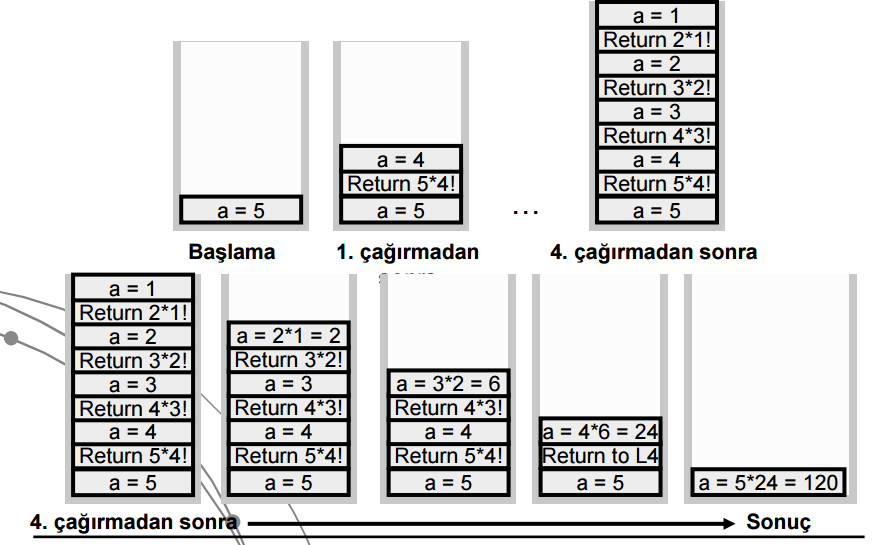
faktor = n \* faktoriyel(n - 1);

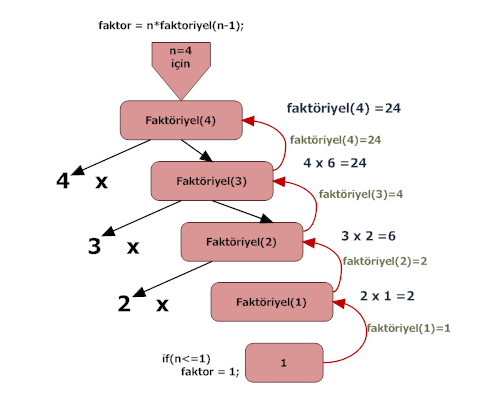


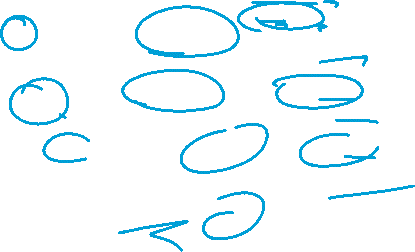
return faktor;



}







#include <stdio.h>

#include <conio.h>

int faktoriyel(int);

int main() {



int sayi;

printf("Sayiyi giriniz : ");



scanf("%d", &sayi);



int fakt = faktoriyel(sayi);



printf("%d! = %d\n", sayi, fakt);



getch();

return 0;

}

int faktoriyel(int n) /\*Foksiyon tanimlari\*/ {

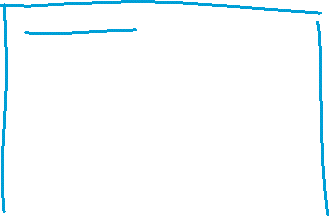
int faktor;

if (n <= 1){

faktor = 1;



printf("1!=1\n");



}



else

{

printf("%d! = %d\*%d!\n",n,n,n-1);

faktor = n \* faktoriyel(n - 1);

}



return faktor;

}



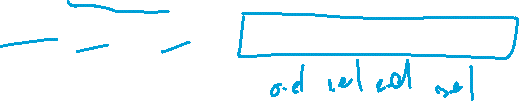
Öz yinelemeli fonksiyonlar kullanmasın avantaj ve dezavantajları nelerdir.???



**ÖRNEK 2: fibonatci sayıları**

0 1 1 2 3 5 8 13 21 …………..

Fibonatci(0)=0 ve Fibonatci(1)=1 dışında Fibonatci(n)=fibonatci(n-1)+ fibonatci(n-2)



#include <stdio.h>

#include <conio.h>



int fibonacci(int);



int main() {



int sayi;

int sonuc;

printf("Sayi giriniz : ");

scanf("%d", &sayi);

sonuc = fibonacci(sayi);

printf("Fibonacci(%d) = %ld", sayi, sonuc);

getch();

return 0;

}

int fibonacci(int n) {



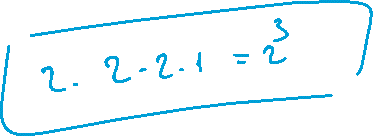
int fib;



if (n <= 1)



fib = n;



else



fib = fibonacci(n - 1) + fibonacci(n - 2);

return fib;

}

**ÖRNEK 3: üst alma**



#include <stdio.h>

#include <conio.h>

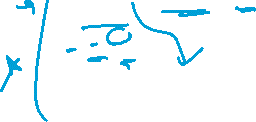


double power(double taban, int us) {

double test;

if (us == 0)

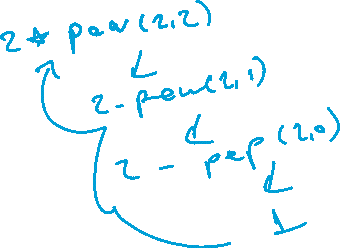
test = 1;



else

test = taban \* power(taban, us - 1);

return test;



}



int main() {

int uss;

double tab, sonuc;

printf("Tabani giriniz : ");

scanf("%lf", &tab);

printf("Ustu giriniz : ");

scanf("%d", &uss);

sonuc = power(tab, uss);

printf("(%lf)^%d = %lf", tab, uss, sonuc);

getch();

return 0;

}

**ÖRNEK 3: palingdrom örneği**



#include <stdio.h>

#include <conio.h>

#include <string.h>

int palindrommu(char \*str, int len)



{



if(len<2) return 1;



return ( str[0] == str[len-1] ) && palindrommu( &str[1] , len-2 );

}



int main() {

char kr,mtn[100];

int sonuc;

do {

printf("Metin giriniz...:");

gets(mtn);

sonuc=palindrommu(mtn,strlen(mtn));

if (sonuc) printf("Metin palingdrom dur.\n");

else printf("Metin palingdrom degildir.\n");

printf("\nDevam etmek istermisiniz...?(e/h)");

kr=getch();

if (kr=='h'||kr=='H') break;

}while(1);

return 0;

}

**ÖDEVLER:**

1. Kendisine parametre olarak gönderilen iki tam sayının OBEB (Ortak Bölenlerin En Büyüğü) ini bulan **öz yinelemeli fonksiyonları** yazınız.
2. Girilen sayıya kadar olan pozitif tamsayıların toplamını bulan **öz yinelemeli** fonksiyonssu yazınız.