

İskenderun Teknik Üniversitesi



Fakülte: Mühendislik ve Doğa Bilimleri *Fakültesi*

Bölüm: *Bilgisayar Mühendisliği Bölümü*

Ders: Algoritmalar ve Programlama

Dönem: 2021 – 2022 (Güz)

Öğretim üyesi: *Dr. Öğr. Üyesi Mehmet SARIGÜL*

10.

Hafta

- Tartıştığımız programlar genellikle disiplinli, hiyerarşik bir şekilde birbirini çağıran işlevler olarak yapılandırılmıştır.
- Bazı problem türleri için, işlevlerin kendilerini çağırması yararlıdır.
- Özyinelemeli işlev, kendisini başka bir işlev aracılığıyla doğrudan veya dolaylı olarak çağıran bir işlevdir.
- Özyineleme, üst düzey bilgisayar bilimleri derslerinde ayrıntılı olarak tartışılan karmaşık bir konudur.
- Bu bölümde ve sonraki bölümde, basit özyineleme örnekleri sunulmaktadır.

Özyineleme

- Yinelemeli problem çözme yaklaşımlarının birçok ortak noktası vardır.
- Bir sorunu çözmek için özyinelemeli bir işlev çağrılır.
- İşlev aslında yalnızca en basit durum (lar) 1 veya sözde temel durum (lar) 1 nasıl çözeceğini bilir.
- İşlev bir temel durumla çağrılırsa, işlev basitçe bir sonuç verir.
- Fonksiyon daha karmaşık bir problemle çağrılırsa, fonksiyon problemi iki kavramsal parçaya ayırır: fonksiyonun nasıl yapılacağını bildiği bir parça ve nasıl yapılacağını bilmediği bir parça.

Özyineleme

- Özyinelemeyi mümkün kılmak için, ikinci parça orijinal probleme benzemeli, ancak biraz daha basit veya daha küçük bir versiyon olmalıdır.
- Bu yeni sorun orijinal soruna benzediğinden, işlev daha küçük sorun üzerinde çalışmak üzere kendisinin yeni bir kopyasını başlatır (çağırır) - buna özyinelemeli çağrı veya özyineleme adımı denir.
- Özyineleme adımı aynı zamanda dönüş anahtar sözcüğünü de içerir, çünkü sonucu, işlevin orijinal arayana geri gönderilecek bir sonuç oluşturmak için nasıl çözüleceğini bildiği problem bölümü ile birleştirilecektir.
- Özyineleme adımı, işleve yönelik orijinal çağrı duraklatıldığında, özyineleme adımının sonucunu beklerken yürütülür.

Özyineleme

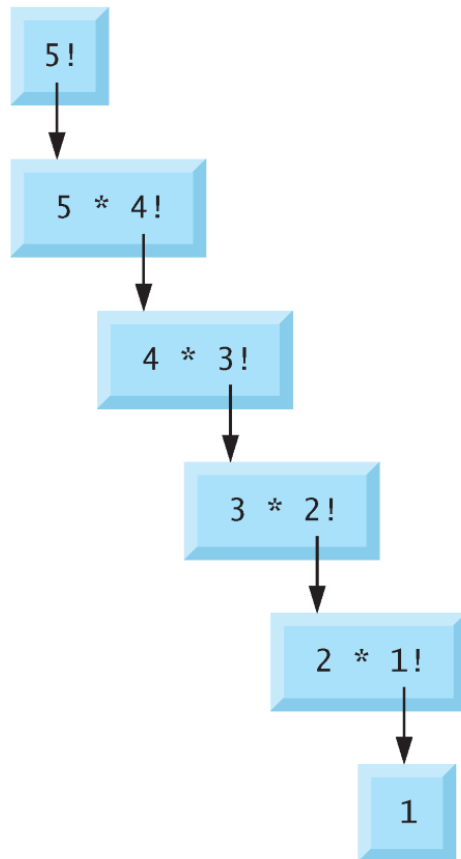
- İşlev, çağrıldığı her sorunu iki kavramsal parçaya ayırmaya devam ettiği için özyineleme adımı, bu tür daha birçok özyinelemeli çağrıya neden olabilir.
- Özyinelemenin sona ermesi için, işlev kendisini orijinal sorunun biraz daha basit bir versiyonuyla her çağırdığında, bu daha küçük problemler dizisi sonunda temel durumda birleşmelidir.
- İşlev temel durumu tanıdığı anda, işlevin önceki kopyasına bir sonuç döndürür ve işlevin orijinal çağrısı nihai sonucu ana sonuca döndürene kadar satırın sonuna kadar bir geri dönüş dizisi başlar.

Özyineleme

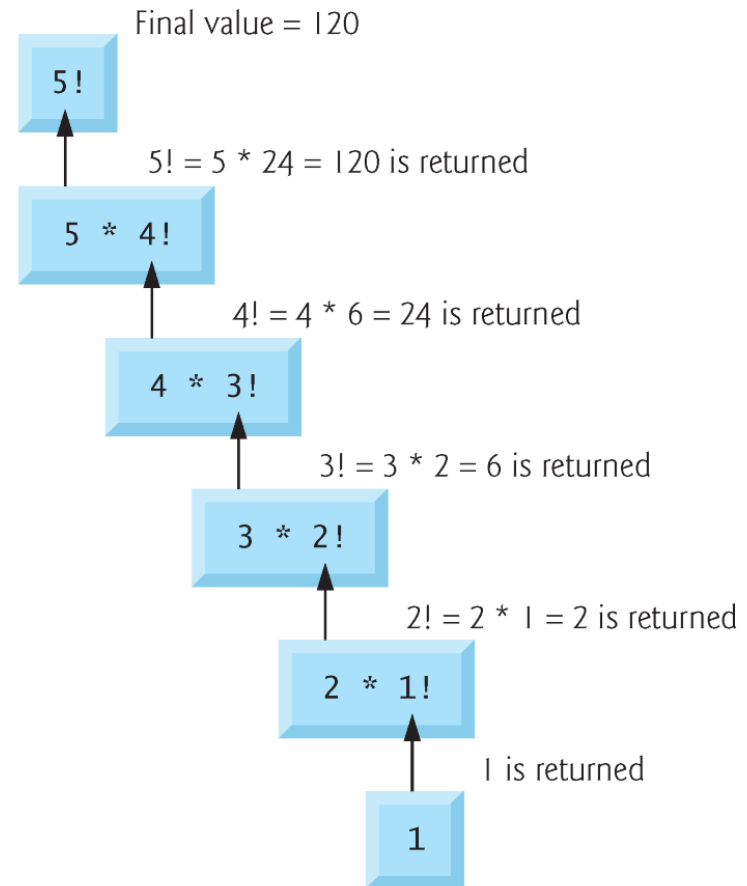
- Faktörleri Yinelemeli Olarak Hesaplama
 - Negatif olmayan n tamsayısının faktöriyeli, $n!$ ("n faktöriyel" olarak okunur), çarpım
 - $n \cdot (n - 1) \cdot (n - 2) \cdot \dots \cdot 1$
 - 1 ile! 1'e eşit ve $0!$ 1 olarak tanımlanmıştır.
 - Faktöriyel işlevin özyinelemeli tanımına aşağıdaki ilişki gözlemlenerek ulaşılır:
 - $n! = n \cdot (n - 1)!$
 - Örneğin, $5!$ açıkça $5 \cdot 4!$ 'e eşittir! aşağıda gösterildiği gibi:
 - $5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ $5! = 5 \cdot (4 \cdot 3 \cdot 2 \cdot 1)$ $5! = 5 \cdot (4!)$

Özyineleme

a) Sequence of recursive calls



b) Values returned from each recursive call



- `#include <stdio.h>`
- `using namespace std;`
- `int fact(int i)`
 - `{`
 - `if (i == 0 || i == 1)`
 - `return 1;`
 - `else`
 - `return i * fact(i - 1);`
 - `}`
- `int main()`
 - `{`
 - `for (int i = 0; i <= 10; i++)`
 - `printf(«%d», fact(i));`
 -
 - `}`

- Fibonacci özyinelemeli bir problemdir. Özyineleme ile fibonacci dizisini yazdırınız.
- $f(0)=2$, $f(1)=5$ veriliyor. $f(x)=f(x-1)*2+f(x-2)/2$ ise verilen x değerine göre $f(x)$ i bulan programı özyinelemeli ve özyinelemesiz yazınız.

Ödev





Ders Sonu

