

Programlama Dilleri (315)

Dr. Öğr. Üyesi Ahmet Arif AYDIN

Abstract Data Types and Encapsulation Structures
Soyut Veri Tipleri ve Kapsülleme Yapıları

- Altprogramların Gerçekleştirilmesi
- Call
- Return
- Faktöriyel

Abstraction (Soyutlama)

- Bir varlığın en önemli öğelerini içeren genel görünüş soyutlama olarak isimlendirilir. üst sınıf daha genel, ortak ve abstract bir görüntü oluşturur.
- Varlıkların ortak özelliklerini bir grup altında toplamak için kullanılmaktadır.
- Örneğin kara araçları çeşitlerinin hepsinin genel özellikleri abstraction yardımıyla tanımlanabilir.
 - Motor
 - Yakıt
 - Teker

Abstraction (Soyutlama)

- Programlama işlemi kolaylaştırmak için abstraction kullanılmaktadır.
1. Process Abstraction (süreç-işlem soyutlama)
 - Altprogramlar birer soyutlamadır
 - Method ve fonksiyon çağrıları
 - Sırala (dizi, artan veya azalan)
 2. Data Abstraction (veri soyutlama)
 - Records

Veri Soyutlama (Data Abstraction)

- Kayıt tipleri (records) veri soyutlama yapılarıdır.
- Kullanıcı tanımlı veri tipidir

```
struct NameType {  
    char first[15];  
    char middleInit;  
    char last[15];  
};
```

```
struct StudentType {  
    NameType name;  
    int      idNum;  
    float    credits;  
    float    gpa;  
};
```

Soyut veri tipi detayları gizlemektedir.

Bir soyut veri tipinin örneği bir nesnedir (object)

Veri Soyutlama (Data Abstraction)

- Floating-point
 - Kayan noktalı sayılar üzerinde yapılan işlemler programlama dili tarafından tanımlanmıştır.
 - İşlemler gerçekleşirken bellek üzerinde soyutlanan ek bir hücre kullanılmaktadır
- Kullanıcı tanımlı soyut veri tipleri
 - Detayları gizlenebilen tip tanımlama imkanı sağlamalı
 - Tanımlana tip üzerinde bir dizi işlem gerçekleştirebilmeli

Soyut Veri Tipleri (Abstract Data Types)

- C++ da class ve struct yapıları soyut veri tipleri oluşturmayı sağlar
- Struct yapısı veri eklendiğinde kullanılır.

Soyut Veri Tipleri (Abstract Data Types)

class yapısı

- Encapsulation (kapsulleme)
 - Veri (data members)
 - Methodlar (member functions)
 - Class da veya instance (aynı metodlar - farklı veri)
 - Instance (statik veya dinamik) (new (oluştur) , delete (sil))
- Information Hiding (bilgi gizleme)
 - Public, private

Soyut Veri Tipleri (Abstract Data Types)

class yapısı

- Constructors : Class tanımının içinde yer alır ve bazı değişkenlere ilk değer atamada kullanılır
- Destructors: class örneğinin yaşam siresi bittiginde örtülü olarak çağrılır.

Soyut Veri Tipleri (Abstract Data Types)

```
1  #include <iostream>
2  using namespace std;
3  class Adder {
4      public:
5          // constructor
6          Adder(int i = 0) {
7              total = i;
8          }
9          // dışarıdan erişilebilen method
10         void addNum(int number) {
11             total += number;
12         }
13
14         // dışarıdan erişilebilen method
15         int getTotal() {
16             return total;
17         };
18     private:
19         // dışarıdan gizlenen değişken
20         int total;
21 };
22 int main() {
23     Adder a;
24     a.addNum(10);
25     a.addNum(20);
26     a.addNum(30);
27
28     cout << "Total " << a.getTotal() << endl;
29     return 0;
30 }
```

Total 60

Kapsulleme Yapıları (Encapsulation Constructs)

- `using namespace std;` (C++)
- `package k;` (Java)
- `import k.myCode;` (Java)
- `require 'myStuffMod'` (ruby)