

Programlama Dilleri Laboratuvar Föyü

C++ Programlama Dili 1

Dil Yapıları ve Standart Şablon Kütüphanesi

Şevket Umut ÇAKIR

1 Giriş

C++ düşük seviyeli işlemlerin yapılması ve hız gerektiren uygulamalar için tercih edilen nesneye yönelik bir programlama dilidir. C++ başlangıçta 1979 yılında Bjarne Stroustrup tarafından C dilinin genişletilmiş hali olarak geliştirilmiştir. C dilinde yazılan bir çok program kaynak kodunu değiştirmeden C++ derleyicilerinde derlenebilir. C++'ın C++98, C++03, C++11, C++14, C++17 ve C++20 gibi standartları bulunmaktadır.

1.1 Kurulum ve IDE Seçimi

Piyasada kullanılabilecek bir çok geliştirme ortamı bulunmaktadır. Bu geliştirme ortamlarından bazıları aşağıda listelenmiştir. Geliştirme ortamlarının yanı sıra ücretli veya ücretsiz olarak kullanılabilecek bir çok C++ derleyicisi bulunmaktadır[1].

- CLion: JetBrains firmasının geliştirme ortamı
- MonoDevelop: Mono projesinin geliştirme ortamı
- C++ Builder: Kolay GUI uygulamaları geliştirilir
- Dev C++: Hafif ve güçlü geliştirme ortamı
- Code::Blocks
- Eclipse, Visual Studio Code, CodeLite, NetBeans

2 Konu Anlatımı ve Deney Hazırlığı

C++ programları C programlarına benzer özellikler gösterir. Aşağıda örnek bir C++ kodu verilmiştir.

```
#include<iostream>

int main() {
    std::cout << "Merhaba dünya!" << std::endl;
    return 0;
}
```

Bu kod ekrana “Merhaba dünya!” yazdırmaktadır. İlk satırdaki `#include<iostream>` ifadesi bir çok C++ programı için ortak olarak kullanılan standart girdi/çıkı işlemlerinin yapılması içindir. `cout << "Merhaba dünya!" << endl;` ifadesi `cout` komutu ile ekrana ilgili ifadeyi yazdırır. `<<` işleci akış yönlendirmek için kullanılır ve `cout` ile kullanılarak birden fazla elemanın yazdırılması sağlanır. Ad alanı(namespace) dil öğelerini gruplamak için kullanılır (Java dilindeki package yapısı gibi). Aşağıda aynı örneğin farklı bir hali vardır. İkinci satırdaki ifade ile `std` ad alanını ilgili yapıların başına eklemeye gerek kalmaz.

```
#include<iostream>
using namespace std;
int main() {
    cout << "Merhaba dünya!" << endl;
    return 0;
}
```

2.1 Veri Türleri

C++ dilinde C dilindeki `int`, `double`, `float`, `byte` vb. veri tipleri bulunmaktadır. Bu tiplerin yanı sıra çeşitli sınıflar da bulunmaktadır. Örneğin `string` çok kullanılan bir veri tipidir.

2.2 Giriş/Çıkış İşlemleri

C++ dilinde standart girdi işlemleri `cin` ve standart çıktı işlemleri de `cout` ile gerçekleştirilir. Aşağıda örnek kullanımları vardır.

```
#include<iostream>
using namespace std;
int main(int argc, char* argv[])
{
    double boy, kilo;
    cout<<"Boyunuzu girin(örn: 1.90): ";
    cin>>boy;
    cout<<"Kilonuzu girin(örn: 85):";
    cin>>kilo;
    double vki=kilo/(boy*boy);
    cout << "Vücut Kitle Endeksi: " << vki << endl;
    if(vki <= 18.5)
        cout << "Zayıf" << endl;
}
```

```

else if (vki <= 25)
    cout << "Normal" << endl;
else if (vki <= 30)
    cout << "Kilolu" << endl;
else
    cout << "Aşırı kilolu" << endl;
return 0;
}

```

2.3 Standart Şablon Kütüphanesi

Standart şablon kütüphanesi içinde algoritmalar, konteynırlar, fonksiyonlar ve iteratörlerin bulunduğu bir kütüphanedir. Bu kütüphanede C++ dilinde polimorfik olarak kullanılabilecek sınıflar da bulunmaktadır. Java dilindeki ArrayList yapısına benzer olan Vector yapısı çok kullanılmaktadır. Sık kullanılan bazı vector fonksiyonları aşağıda verilmiştir.

fonksiyon	işlev	fonksiyon	işlev
begin	Başlangıç(iterator)	end	Bitiş(iterator)
size	Boyutunu verir	at(i)	i indisindeki elemanı verir
front	İlk eleman	back	Son eleman
push_back	Sona eleman ekler	pop_back	Sondan eleman siler
erase	Belirtilen konum veya aralıktaki elemanları siler	clear	Tüm elemanları siler

Aşağıda vector yapısının kullanımına örnek verilmiştir.

```

#include <iostream>
#include <vector>
using namespace std;

void yazdir(vector<int> v) {
    for (auto iter = v.begin(); iter != v.end(); ++iter)
    {
        cout << *iter << " ";
    }
    cout << endl;
}

void yazdir2(vector<int> v) {
    for (int i = 0; i < v.size(); ++i)
    {
        cout << v[i] << " ";
    }
    cout << endl;
}

```

```

int main(int argc, char* argv[])
{
    vector<int> sayilar;
    sayilar.push_back(3);
    sayilar.push_back(5);
    sayilar.push_back(7);
    sayilar.push_back(9);
    sayilar.insert(sayilar.begin(), 11);
    sayilar.insert(sayilar.begin()+2, 13);
    yazdir(sayilar);
    sayilar.erase(sayilar.begin(), sayilar.begin()+2);
    yazdir(sayilar);
    return 0;
}

```

3 Deneyin Uygulanması

3.1 Taş, Kağıt ve Makas

Bu uygulamada iki kişi ile oynanan klasik taş, kağıt, makas oyununu gerçekleştirmek istenmektedir. Kullanıcıdan içerikleri {"taş", "kağıt", "makas"} olan iki adet metin alınacaktır. İlk girilen metin birinci oyuncunun seçimini, ikinci girilen metin de diğer oyuncunun seçimini göstermektedir. Girilen metinlere göre ekrana "birinci oyuncu kazandı", "ikinci oyuncu kazandı" veya "berabere" mesajlarını yazdıran C++ kodunu yazınız.

3.2 Mükemmel Sayı

Tam bölenlerinin toplamı kendisine eşit olan sayılara mükemmel sayı denir. 28 ve 96 mükemmel sayılara verilebilecek iki örnektir. Kullanıcıdan alına sayının mükemmel sayı olup olmadığını bulan, mükemmelse ekrana "mükemmel" aksi takdirde "mükemmel değil" yazan C++ kodunu yazınız.

3.3 Collatz Sanısı

Henüz doğruluğu kanıtlanmamış olan Collatz sanısı yapılan iki işlem sonunda 1'e indirgenebileceğini söyler. Bu iki işlem aşağıdaki gibidir:

$$f(n) = \begin{cases} \frac{n}{2}, & n \text{ çift ise} \\ 3n + 1, & n \text{ tek ise} \end{cases}$$

$$a_i = \begin{cases} n, & i=0 \\ f(a_{i-1}), & i>0 \end{cases}$$

Collatz sanısında bir seri sayının kendisi ile başlar ve 1 ile sonlanır. Örneğin 13 sayısı için Collatz serisi 13, 40, 20, 10, 5, 16, 8, 4, 2,1 sayılarından oluşan 10 sayılık bir seridir.

Kullanıcının girdiği herhangi bir sayının Collatz serisinde kaç adımda 1'e ulaştığını bulan C++ programını yazınız.

3.4 Pascal Üçgeni

Pascal üçgeni, matematikte binom katsayılarını içeren üçgensel bir dizidir. Pascal üçgenindeki değerler sola doğru yaslanırsa aşağıdaki şekildekine benzer bir matris oluşur. Parametre olarak verilen n değerine göre $n \times n$ lik Pascal üçgenini içeren matrisi hesaplayıp döndüren `pascal` fonksiyonunu C++ dilinde yazınız. Fonksiyona parametre olarak içinde `vector` nesnelerinin bulunduğu bir `vector` nesnesi gönderilmektedir. Gelen parametre bir matrisi temsil etmektedir.

1			
1	1		
1	2	1	
1	3	3	1

Tablo 1: 4 satırlı Pascal üçgeni

Kaynaklar

- [1] *C++ Derleyicileri*. URL: <http://www.compilers.net/Dir/Free/Compilers/CCpp.htm> (son erişim: 2.4.2019).