

5.Ders : C switch ... case

Bu derste, C programlamada bir switch ifadesi yazmayı öğreneceksiniz (bir örnek ile).

If ve else..if ikilisi, birçok alternatif arasında bir blok kodu yürütmenize izin verir.

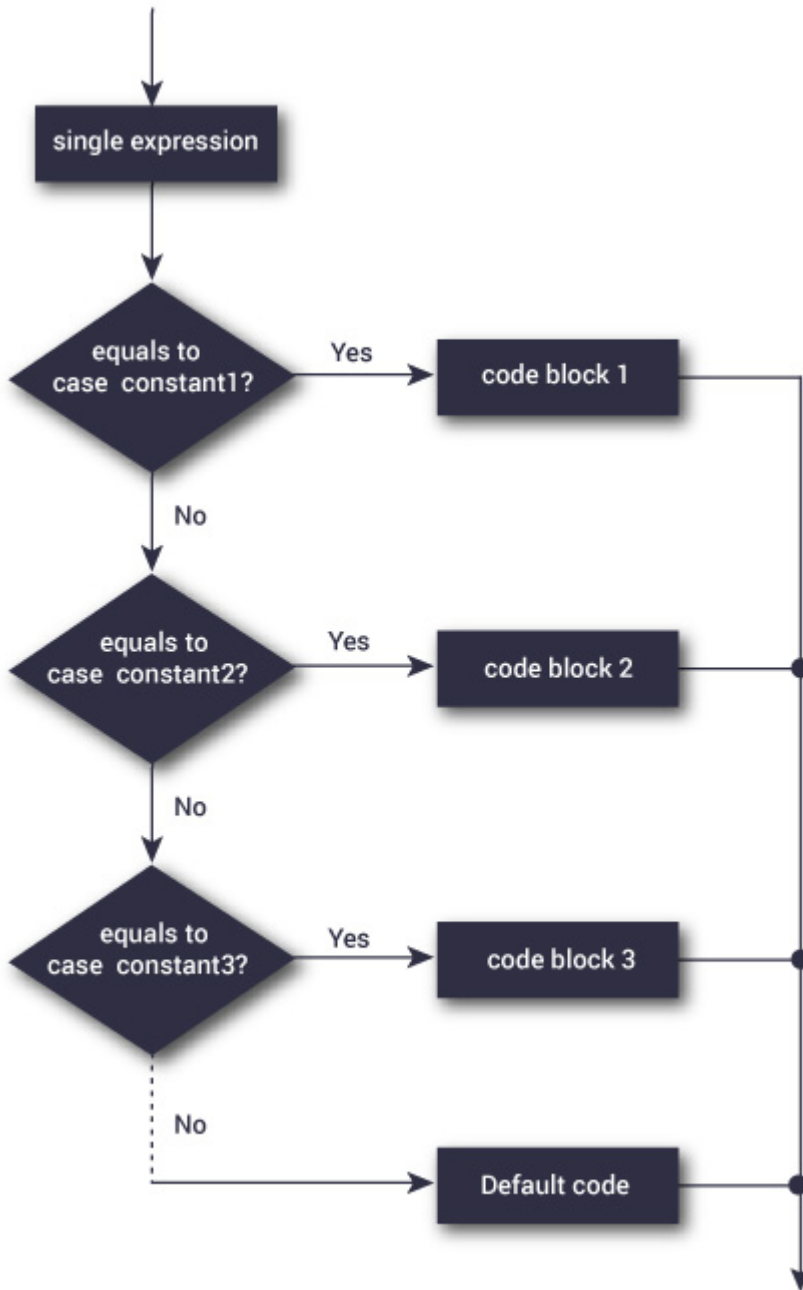
Ancak if deyimi genellikle iç içe olduğundan **Switch deyimi** daha hızlıdır. Ayrıca, switch ifadesinin sözdizimi daha temiz ve anlaşılması kolaydır.

```
switch (n)
{
    case constant1:
        // n constant1 'e eşit ise çalıştırılacak kod;
        break;

    case constant2:
        // n constant2'ye eşit ise çalıştırılacak kod;
        break;
    .
    .
    .
    default:
        // n herhangi bir sabitle eşleşmezse çalıştırılacak
kod
}
```

Switch ifadesiyle eşleşen bir sabit bulunduğunda, programın kontrolü bu durumla ilişkili kod bloğuna geçer.

Farz edelim ki n değeri `constant2`'ye eşittir. Derleyici, `constant2`: ifadesinden sonra `break` ile karşılaşılan ifadeleri çalıştırır. Break ifadesiyle karşılaşıldığında, **switch** ifadesi sonlandırılır.



Örnek: switch ifadesi

```
// switch ifadesini kullanarak Basit bir hesap makinesi yazalım
```

```
#include <stdio.h>

int main() {

    char operator;
    double firstNumber,secondNumber;

    printf("bir işlem seçin (+, -, *, /): ");
    scanf("%c", &operator);

    printf("iki sayı giriniz: ");
    scanf("%lf %lf",&firstNumber, &secondNumber);

    switch(operator)
    {
        case '+':
            printf("%.1lf + %.1lf = %.1lf",firstNumber,
secondNumber, firstNumber+secondNumber);
            break;

        case '-':
            printf("%.1lf - %.1lf = %.1lf",firstNumber,
secondNumber, firstNumber-secondNumber);
            break;

        case '*':
            printf("%.1lf * %.1lf = %.1lf",firstNumber,
secondNumber, firstNumber*secondNumber);
            break;

        case '/':
            printf("%.1lf / %.1lf = %.1lf",firstNumber,
secondNumber, firstNumber/secondNumber);
            break;

        // operatör herhangi bir durumla eşleşmiyor (+, -, *,
/)
        default:
            printf("Hata! bir sayı girmediniz");
    }
}
```

```
    return 0;  
}
```

Çıktısı :

```
bir işlem seçin (+, -, *,): -  
iki sayı giriniz: 32.5  
12.4  
32.5 - 12.4 = 20.1
```

Kullanıcı tarafından girilen – operatörü operatör değişkeninde saklanır. Ve iki işlenen 32.5 ve 12.4 sırasıyla firstNumber ve secondNumber değişkenlerinde saklanır.

```
printf("%.1lf / %.1lf = %.1lf",firstNumber, secondNumber,  
firstNumber/firstNumber);
```

Son olarak, **break** ifadesi **switch** ifadesini sonlandırır