**Algoritma Konusu:** Algoritma Tamamlama ve Algoritma Oluşturma Etkinliği

* **Önemli Bilgiler**   
  Çeşitli görevleri gerçekleştirmek amacıyla hazırlamış programlara **yazılım** adı verilir.
* Her yazılım bir **problemi** çözmek amacıyla geliştirilmiştir.
* Bir problemi **çözmek** için
  + **1.** Problemi iyi anla
  + **2.** Çözüm yollarını bulmak
  + **3.** Çözüm için yolları adım adım kullan.

|  |
| --- |
| * Bilgisayarlar da problemleri tıpkı bizler gibi  çözmeye çalışır. Kullanıcı tarafından  kendisine verilen komutları **adım adım**  uygulayarak problemin çözümüne ulaşır. * Kullandığımız yazılımların tamamı **«kod»** adı verilen bilgisayarın anlayacağı dilde yazılmış özel komutlardan oluşur. * Kodlamaya başlamadan problemin çözümünün adım adım gösterilmesine **Algoritma** denir. |
| **Bir Problemi Çözerken** |
| **Dikkat et**   * Her algoritma “Başla” ile başlar “Bitir” ile biter * İşlemler sırayla numaralandırılarak yazılır * Algoritma çizerken sıraya bakılarak şekiller oklarla birleştirilir * Kontrol et şekli kullanılırken sonuç 2 tane olur “Evet ya da Hayır” |

|  |  |
| --- | --- |
| **Akış Şeması Şekilleri** | **Anlamları(Yazınız)** |
|  |  |
|  | . |
|  |  |
|  |  |
|  |  |

**Algoritma ve Akış Diyagramının Faydaları**

* Program yazmayı kolaylaştırır.
* 2.Hatalı kodlama oranını azaltır.
* 3.Program yazımı için geçen süreyi kısaltır.
* 4. İşlem akışını açık bir şekilde gösterdiğinden programı kontrol etmek kolaylaşır.
* 5. Sonradan yapılacak düzenlemelerde kolaylıklar sağlar.

**Algoritma Hazırlama Adımları**

* Önce algoritmada kullanılacak değişken, sabit ve sayaç isimleri belirlenir.
* Varsa veri girişleri yapılır.
* Yapılacak işlem ve formüller yazılır.
* Sonuçlar ekrana yazdırılır veya çevre birimlerine gönderilir.

**Algoritmalarda Kullanılan Terimler**

**Sabit:** Programdaki değeri değişmeyen ifadelere “sabit” denir.

**Değişken:** Programın her çalıştırılmasında, farklı değerler alan bilgi/bellek alanlarıdır.

**Aktarma:** Herhangi bir bilgi alanına veri yazma, herhangi bir değişkenin sonucunu başka bir değişkende gösterme vb. görevlerde aktarma operatörü (=) kullanılır.

**Program:** Belirli bir işi gerçekleştirmek için gerekli komutlar dizisi olarak tanımlanabilir.

**Programlama:** Bir programı oluşturabilmek için gerekli komutların belirlenmesi ve uygun biçimde kullanılmasıdır.

**ALIŞTIRMA:**

**Günlük hayatımızda karşılaştığımız 5 tane problem ve bu problemleri çözecek algoritmaları yazalım.**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**PEKİ BİLGİSAYARDA ÇÖZÜLECEK BİR SORUNU NASIL ALGORİTMA İLE İFADE EDERİZ?**

Bunun için öncelikle bir sorun tanımlayalım. Başlangıç ta basit olması için şöyle bir problem üzerinde düşünelim. Bilgisayara verilecek ***iki sayıyı toplayıp sonucu ekrana yazacak*** bir program için algoritma geliştirmek isteyelim. Sorun son derece basit ancak sistem tasarımının net yapılabilmesi için sorun hakkında anlaşılamayan tüm belirsiz noktalar açıklığa kavuşturulmalıdır.

*1. BAŞLA*

*2. A sayısını oku*

*3. B sayısını oku*

*4. TOPLAM=A + B işlemini yap*

*5. TOPLAM değerini ekrana yaz*

*6. SON*

ŞİMDİ BU İŞLEMİ AKIŞ DİYAGRAMI İLE GÖSTERELİM

|  |
| --- |
|  |

**ÖRNEK 2:**

Aynı algoritmayı çarpma işlemi için de kullanalım. Algoritmayı yazınız.

|  |
| --- |
| 1.  2.  3.  4.  5.  6. |

!ŞİMDİ BU İŞLEMİ AKIŞ DİYAGRAMI İLE GÖSTERELİM

|  |
| --- |
|  |

**ÖRNEK 3:**

Klavyeden girilen iki sayıdan büyük olanı bulup ekranda yazdıran algoritmayı yazalım.

|  |
| --- |
| 1.  2.  3.  4.  5.  6.  7.  8. |