

YMH213 MESLEKİ İNGİLİZCE

Dr. Öğr. Üyesi Bihter DAŞ

Algorithms and Flowcharts

(Algoritmalar ve Akış Şemaları)

Algorithm means «putting forward a specific task step by step by using pre-defined processing steps, and to encode these steps in computer environment by using any programming language». In other aspect, algorithm is «logical and symbolic description of processes required for the solution of a problem».

Algoritma, «önceden tanımlanmış işlem adımlarını kullanarak belirli bir görevi adım adım ileriye doğru yürütmek ve bu adımları herhangi bir programlama dili kullanarak bilgisayar ortamında kodlamak» demektir. Diğer bir açıdan, algoritma, «bir problemin çözümü için gerekli süreçlerin mantıksal ve sembolik açıklamasıdır».

Algorithms and Flowcharts (Algoritmalar ve Akış Şemaları)

An Algorithm puts forward the steps and conditions to be followed in order to get the results of a specific job or problem. These steps can reach a conclusion if the conditions are followed step by step. In computer application, many algorithms are required while developing software applications.

Bir Algoritma, belirli bir işin veya problemin sonuçlarını almak için takip edilen adımları ve koşulları ortaya koyar. Bu adımlar, koşullar adım adım takip edildiğinde bir sonuca varabilir. Bilgisayar uygulamasında, yazılım uygulamaları geliştirilirken birçok algoritmaya ihtiyaç duyulmaktadır.

Necessary Features for an Algorithm (Bir Algoritma İçin Gerekli Özellikler)

To be effective and general: Algorithm should be **efficient** and should not be **repetitious**; it needs to be usable in other algorithms. In addition, it should be designed with a general purpose; in other words, it should get the correct conclusion in each case and in each input value.

Etkili ve genel olma: Algoritma verimli olmalı ve tekrarlı olmamalıdır; diğer algoritmalarda kullanılabilir olması gerekir. Ayrıca genel bir amaç için tasarlanmalıdır; diğer bir deyişle, her durumda ve her girdi değerinde doğru sonuca varmalıdır.

Necessary Features for an Algorithm

(Bir Algoritma İçin Gerekli Özellikler)

To be finite: An algorithm should include strictly finite number of operations and also time for these operations should be finite. Algorithm consists of a certain number of steps; it starts from a start point and has necessarily an end point.

Sonlu olma: Bir algoritma kesin olarak sonlu sayıda işlem içermeli ve bu işlemler için zaman da sonlu olmalıdır. Algoritma belirli sayıda adımdan oluşur; bir başlangıç noktasından başlar ve bir bitiş noktasına sahip olması gerekir.

Necessary Features for an Algorithm

(Bir Algoritma İçin Gerekli Özellikler)

Infallibility: For algorithm, infallibility is the most important criterion. When algorithm is re-executed, same results are obtained for same input values.

Valid Input/Output: An algorithm should have input and output values. Input value is data that algorithm processes in order to produce a result; output value is the result obtained by algorithm.

Necessary Features for an Algorithm (Bir Algoritma İçin Gerekli Özellikler)

Hatasızlık: Algoritma için hatasızlık en önemli kriterdir. Algoritma yeniden yürütüldüğünde, aynı girdi değerleri için aynı sonuçlar elde edilir.

Geçerli Giriş / Çıkış : Bir algoritma giriş ve çıkış değerlerine sahip olmalıdır. Girdi değeri, algoritmanın bir sonuç üretmek için işlediği verilerdir; çıktı değeri, algoritma ile elde edilen sonuçtur.

Necessary Features for an Algorithm

(Bir Algoritma İçin Gerekli Özellikler)

Performance : An algorithm should be designed to offer good performance; repetitions should be avoided. Memory requirement and working time should be as balanced as possible.

Performans: Bir Algoritma, iyi performans sunmak için bir tasarlanmalıdır; tekrarlardan kaçınılmalıdır. Bellek gereksinimi ve çalışma süresi olabildiğince dengeli olmalıdır.

Pseudo Code and Real Code (Sözde Kod ve Gerçek Kod)

Pseudo code means putting forward or defining algorithm by using both programming language and natural language. On the other hand, real code means implementing algorithm by using a programming language, such as C, Java or C++.

Sözde kod, hem programlama dilini hem de doğal dili kullanarak algoritmayı sunmak veya tanımlamak anlamına gelir. Öte yandan, gerçek kod, C, Java veya C ++ gibi bir programlama dili kullanarak algoritma uygulamak anlamına gelir.

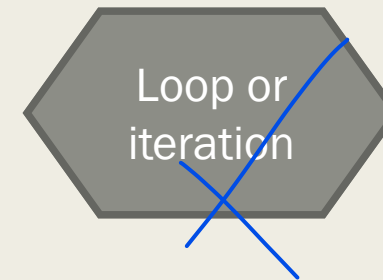
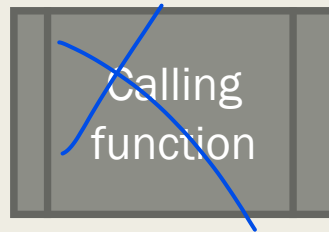
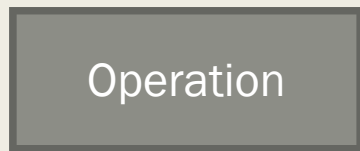
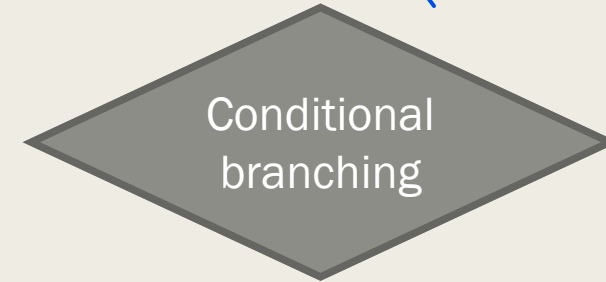
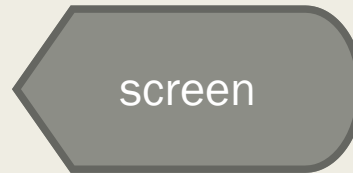
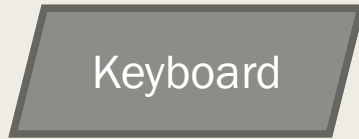
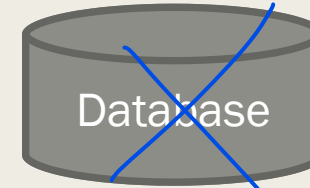
Flowcharts

(Akış çizelgeleri)

Flowcharts are used to put forward the algorithm by explaining the steps that need to be done in a visual way; it shows what should be done to fix the problem from beginning to end by using the symbols that consists of geometric shapes. Each symbol in general indicates a command or task to do. Flowcharts begins with Start icon and ends with Stop icon.

Akış şemaları, yapılması gereken adımları görsel bir şekilde açıklayarak algoritmayı ortaya koymak için kullanılır; Geometrik şekillerden oluşan sembolleri kullanarak sorunun baştan sona giderilmesi için yapılması gerekenleri gösterir. Genel olarak her sembol, yapılacak bir komut veya görevi belirtir. Akış çizelgeleri Başlat simgesiyle başlar ve Durdur simgesiyle biter.

Flowchart Symbols



Loops and Iterative Operations (Döngüler ve Yinelemeli İşlemler)

For example, when reading value to n-element array, it is necessary to write line that consist of n reading commands. To make such repetitive operations, it is more **reasonable** to use loops in algorithms.

Örneğin, n elemanlı diziye değer okurken n okuma komutundan oluşan bir satır yazmak gerekir. Bu tekrarlayan işlemleri yapmak için, algoritmalarda döngü kullanmak daha mantıklıdır.

Algorithm Design with an Object-Oriented Approach

(Nesne Tabanlı Bir Yaklaşımla Algoritma Tasarımı)

Programming languages such as C++ and Java are object-oriented languages; if an object-oriented language is to be used to design a program, flowcharts can be used to express an algorithm. However, it is also possible to use ORM(Object Rule Modeling) or UML(Unified Modeling Language) method which has been developed for object-oriented language.

C ++ ve Java gibi programlama dilleri nesne yönelimli dillerdir; Bir programı tasarlamak için nesne yönelimli bir dil kullanılacaksa, bir algoritmayı ifade etmek için akış şemaları kullanılabilir. Ancak nesne yönelimli dil için geliştirilmiş olan ORM (Object Rule Modeling) veya UML (Unified Modeling Language) yöntemini kullanmak da mümkündür.

Summary (Özet)

Algorithm design could be called as "the initial step of programming". Thus, it should be easily implemented by using a programming language.

Algoritma tasarımı, "programlamanın ilk adımı" olarak adlandırılabilir. Bu nedenle, bir programlama dili kullanılarak kolayca uygulanmalıdır.

Key Questions

- What are the basic features of algorithms?
- Draw a flowchart of an algorithm that reads two numbers entered by keyboard; calculates the arithmetic mean of these two numbers and then display it on the screen.
- Design an algorithm that determines bigger of two numbers typed from keyboard.(Use step 1, Step 2 etc.)
- What is the difference between pseduo-code and real code?

References

- Rifat Çölkesen, Introduction to Computer Engineering
- Doç. Dr. Murat Karabatak, ders notları
- <https://www.pearson.com/>