**PROBLEM ÇÖZME = KODLAMA**

Araştırmacılar, problem çözme becerilerinden yoksun çocuklarda depresyon ve intihar eğilimi riskinin

yüksek olabileceğini ortaya koymuştur. Ayrıca araştırmacılar, çocuğun problem çözme yeteneğini geliştirmenin

zihinsel sağlığı da geliştirebileceğini ifade etmektedir.

Günlük hayatta olduğu gibi programlama derslerinin de başarılı bir şekilde sürdürülebilmesi için, öğrencilerimizin sahip olması gereken özelliklerin başında *“problem çözme yeteneği”* gelir.

Programlamayı öğrenmek birçok öğrenci için zor bir süreçtir. Soyut kavramları anlama, genelleme yapma, öğrendiklerini sonraki konulara aktarabilme ve eleştirel düşünme gibi bir hiyerarşi gerektirir.

Programlama; pratik ve yoğun bir çalışma yaklaşımı gerektirir. Mevcut durumlar için öğrenilen bilgilerin, test edilerek uygulamaya dönüştürülmesi gerekir.

**Problemi Tanımlayın**

Günlük hayatta problem ifadesini genellikle can sıkıcı durumlarda kullanırız. İstenmeyen bir durumla karşılaştığımızda bunun hayatımızı tehdit eden bir hale bürünmeden çözülmesi gerektiği kanaatine vararak çözüm aramaya başlarız. Programlama dünyasında ise problem ifadesi; çözülmesi, geliştirilmesi veya tamamlanması gereken durumları ifade etmektedir. Herhangi bir konuda olmasını istediğimiz durum ile mevcut durum arasındaki her türlü eksiklik, problem olarak görülür ve bunları çözmek bir programcı/kod yazan için en önemli görevdir.

**Problem Çözme Stratejileri**

Bir problemi anladıktan ve analiz ettikten sonra, bir algoritma ile çözüm bulmalısınız. Algoritma; belirli bir sürede, belirli bir veri ile problemi çözmek için takip edilen adımlardır. Genellikle algoritmalar basmakalıp izlenmesi gereken adımlar gibi düşünülür. Sıralama, ortalama hesaplama, faktöriyel hesaplama algoritması gibi... Sabit algoritmaları takip eder ve sonuca ulaşırız. Bilgisayar programlamanın problem çözme aşamasında, her probleme uygun algoritmaları tasarlamalısınız. Daha açık ifade etmek gerekirse problem çözme stratejilerinin farkında olarak bunları programlama problemlerine adım adım uygulayabilmeniz gerekir. Örnek olarak topraktaki nem seviyesine göre bir saksıdaki çiçeği sulamak istediğimizi varsayalım. Aşağıdaki örnekte problem ve problemi çözecek algoritma gösterilmektedir.

Problem: Toprak nem seviyesine göre saksının sulanması

Algoritma:

İşlem1: Başla

İşlem2: Toprağın nem seviyesini ölç

İşlem3: Eğer nem seviyesi 450 değerinin altında ise su pompasını 10 saniye boyunca çalıştır işlem2’ye git

İşlem4: Bitir



**Şekil: Saksının nem seviyesini ölçerek toprağı sulayan sistemin akış şeması**

**ALGORİTMA**

Algoritma; bir problemi çözmek veya bir görevi tamamlamak için adım adım detaylandırılmış talimatlardır.

Programcılar, bilgisayarın bir görevi nasıl yapılacağını tarif

eden algoritmalar yazarlar.

Genel olarak düşündüğümüzde algoritmalar hayatımızın

her yerindedir. Sabah uyanıp okula gelinceye kadar

ki süreçten tutun, akşam uyumadan önce diş fırçalamamıza

kadar her aşamada birçok algoritma, biz farkında

olmadan, beynimiz tarafından oluşturulur, test edilir, uygulanır,

güncellenir ve sonlandırılır. Şekil ’de bir öğrencinin

sabah uyandığında gerçekleştirebileceği görevlerin

basit bir algoritması görülmektedir.

**Kendi Algoritmamızı Yazalım**

Örneğin, ayran hazırlamak.

**Algoritmik Düşünmenin Faydaları**

Algoritmik düşünme veya bir problemi adım adım çözebilme yeteneği özellikle matematik ve fen bilimlerinde önemlidir. Çocuklar farkında olmadan özellikle matematikte algoritmaları kullanırlar. Bir bölme işlemi yaparken, çarpma, çıkarma gibi işlemlerden faydalanırlar. Algoritmik düşünme; çocukların, problemleri parçalara ayırmalarını ve çözüme ulaşabilmek için farklı adımları bir süreç içinde yürütebilmelerini sağlar.

**AKIŞ ŞEMASI**

Akış şemaları, algoritmaların görseller kullanılarak ifade edilmesidir. Bu işlemi yaparken farklı geometrik şekiller kullanılır. Algoritmaları yazarken emir kipi ile kurduğumuz cümlelerden faydalanırız. Ancak akış şeması ile yapılması gereken görevler net bir şekilde ifade edilir ve görseller ile desteklenir.

**NEDEN KODLAMA ÖĞRENİYORUZ?**

Bazı çocuklar geleceğin programcılarıdır. Bazı çocuklar ise becerilerini desteklemek için kodlamanın bazı unsurlarını gerektiren işlere sahip olacaklardır. Kodlama; günümüz ve geleceğimizin insan davranışlarını şekillendiren, alışkanlıklarımızı, hareketlerimizi, işlerimizi belirli önceliklere göre sıralayarak yapmamızı sağlayabilen, makineler ile aramızdaki iletişimi kurabilen yeni bir dildir.

Metin tabanlı programlama dillerini tanımlamak daha kolaydır. Bunlar bilgisayar klavyesini kullanarak yazdığımız ve metin dosyası olarak sakladığımız dillerdir.

Blok tabanlı bir dil genellikle “yazma” eyleminden ziyade sürükle ve bırak işlevini kullanır. Bloklar veya elemanlar üzerinde simge-metin etiketlerini kullanabilir. Diyaloglar ve açılır menü seçenekleri sıklıkla kullanılır.

Scratch ve Blockly gibi blok tabanlı programlama dilleri oldukça popülerdir. Metin tabanlı diller ise profesyonel yazılım geliştiriciler tarafından kullanılan, “gerçek” programlama dili olan C/C++, Python, JavaScript gibi programlama dilleridir.