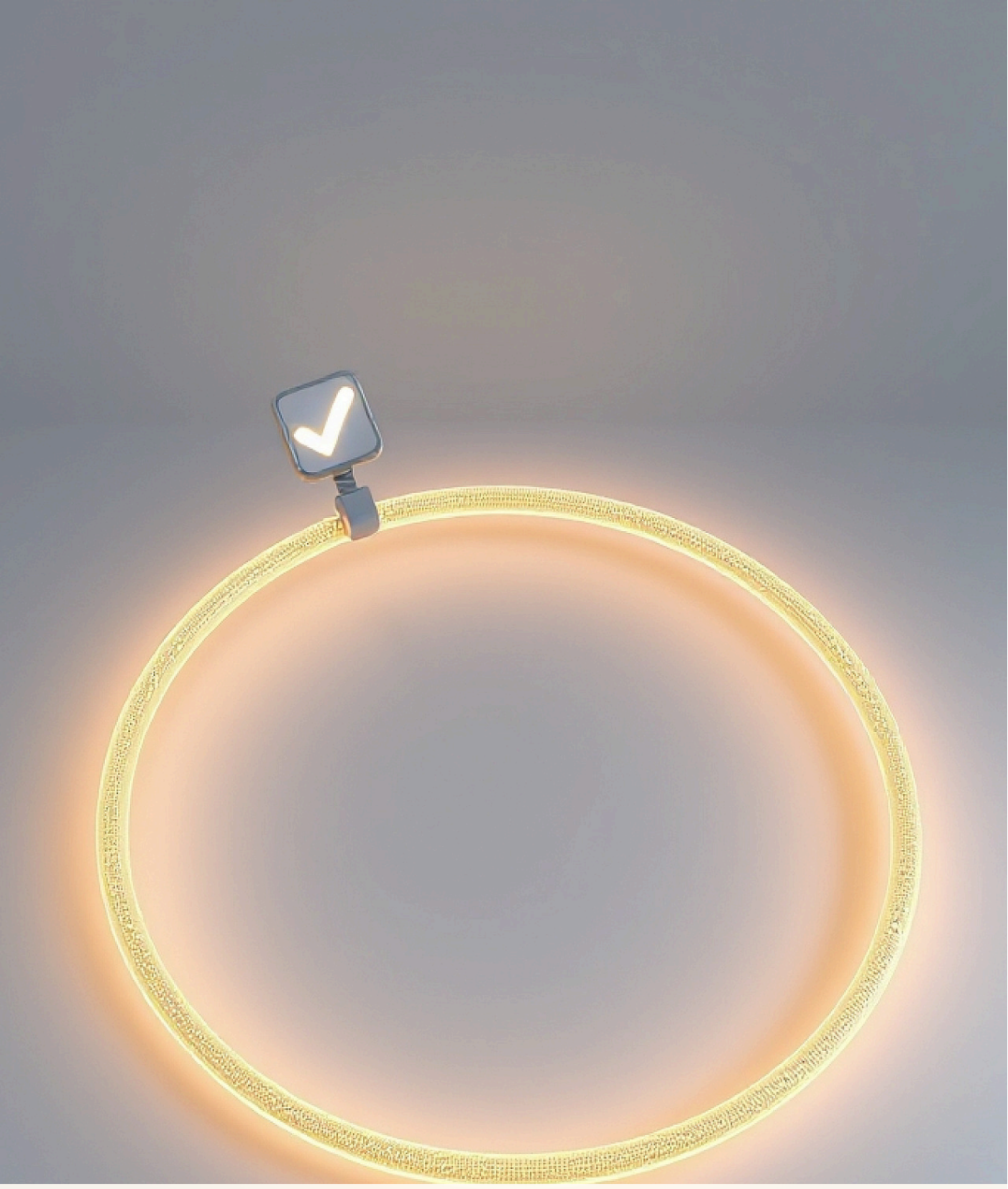
Sonsuz döngü nedir?



www.yzokul.com.tr

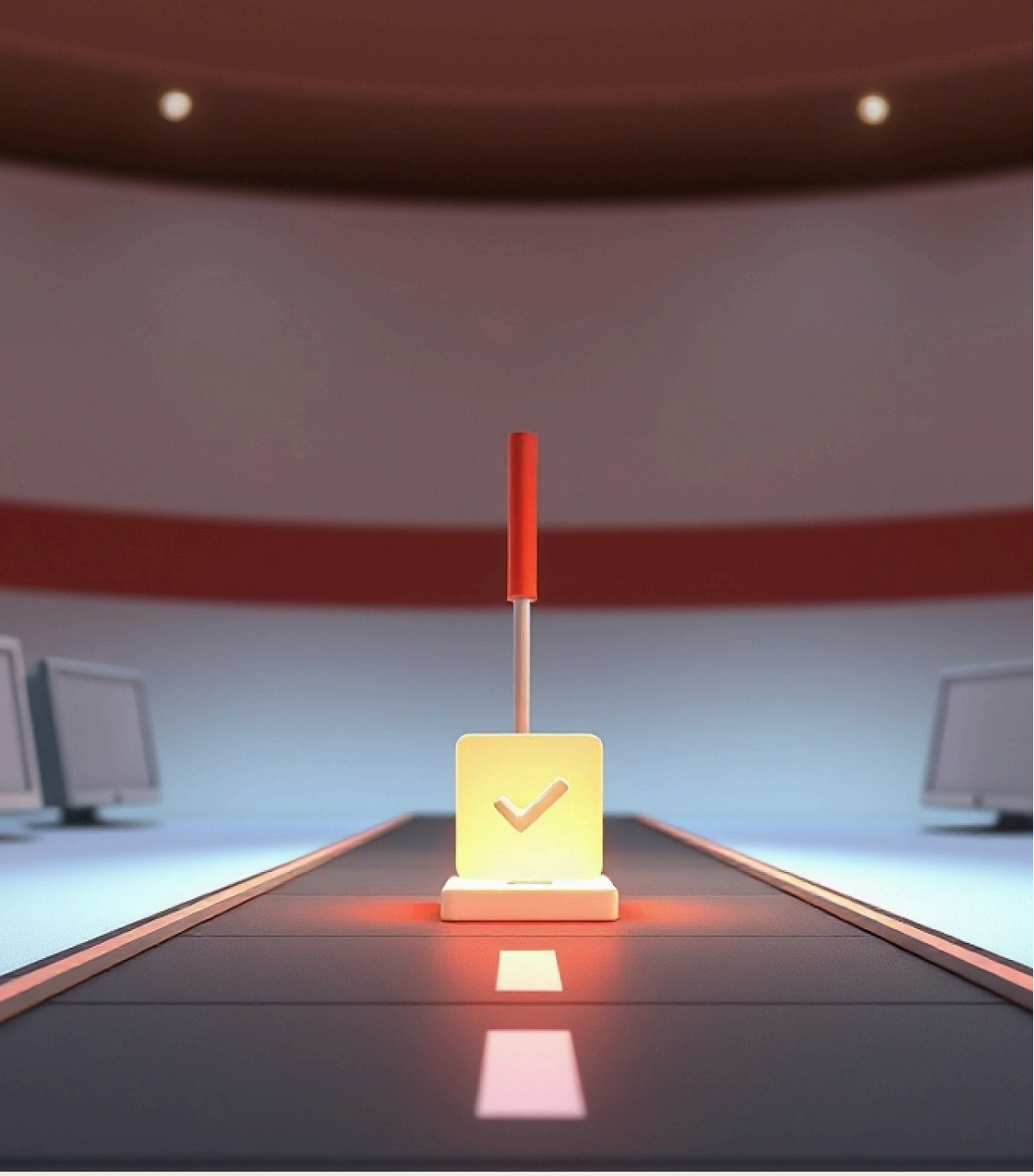
Sonsuz döngü nedir?



Sonsuz döngü nedir? Bitiş koşuluna hiçbir zaman ulaşmayan ve tekrar etmeye devam eden bir döngüdür. Yani içindeki talimatlar durmadan, tekrar tekrar çalışır. Kısa yanıt, sonsuz bir döngünün bitmesi gereken noktaya asla ulaşmadığıdır. İnceleyeceğin temel fikir budur.



Döngü, tekrar tekrar çalışan bir talimatlar kümesidir. Her turda aynı adımlar sırayla yinelenir. Örneğin bir döngü bir sayıya bir ekleyip sonra sayıyı denetleyebilir. Bir şey ona durmasını söyleyene kadar bunu yapmaya devam eder. Döngüler, görevleri yinelemek için programlamada yaygındır.



Her döngünün bir bitiş koşulu vardır; bu, ona ne zaman duracağını söyleyen bir denetimdir. Denetim karşılandığında döngü biter ve program devam eder. Örneğin bir döngü, bir sayı 10'a ulaştığında durabilir. Bitiş koşulu bir bitiş çizgisi gibidir: onu geçtiğinde işin biter. O olmadan döngü ne zaman bırakacağını bilemez.



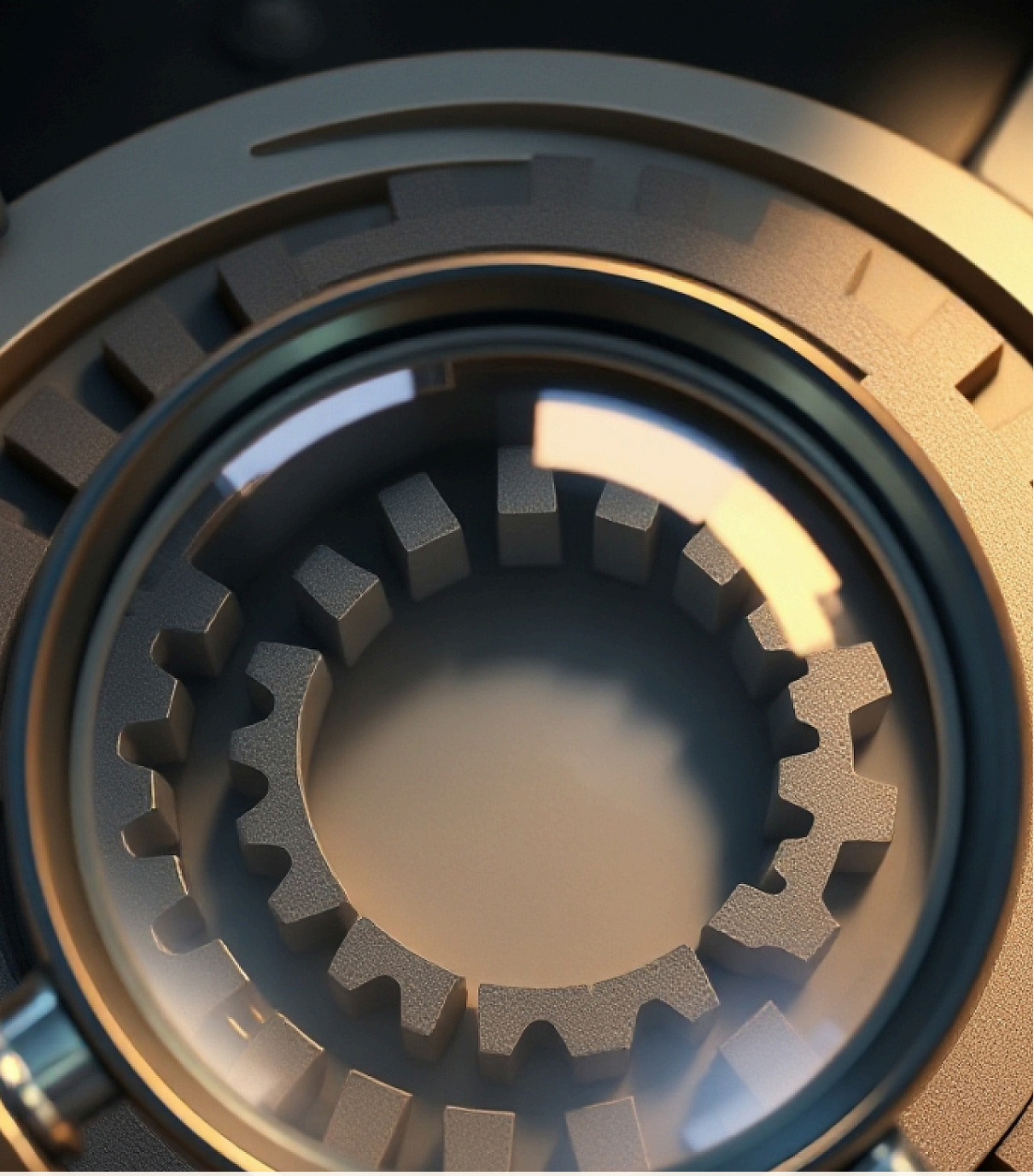
Bitiş koşulu hiçbir zaman doğru olmazsa döngü devam eder. Sonsuz döngünün özü budur. Denetim her seferinde yapılır ama her zaman başarısız olur, bu yüzden döngü asla çıkmaz. Örneğin koşul sayı 10'a eşittir ama sayı hiç değişmiyorsa denetim asla geçmez. Döngü sonsuza kadar çalışır.



Bir sayının 10'dan küçük olup olmadığını denetleyen bir döngü düşün. İçinde her seferinde sayıya bir eklemesi gerekir. Ama bir ekleyen adım eksikse sayı aynı kalır. Böylece sayı < 10 denetimi her zaman doğru olur ve döngü asla bitmez. Bu, sonsuz döngüye yol açan basit bir hatadır.



Hata, döngünün denetlediği değeri güncellemeyi unutması yüzünden oluşur. Bu değişiklik olmadan hiçbir şey bitiş koşuluna doğru ilerlemez. Döngü aynı değerle aynı denetimi yalnızca yineler. Böylece koşul asla yanlış olmaz. İlerleme kaydetmek ve sonunda durmak için değeri güncellemek şarttır.



Eksik değişikliği fark etmek için döngüye bak ve sayıyı değiştirmesi gereken adımı bul. Bir sayma döngüsünde bu adım genellikle bir ekleme gibi bir artırmadır. Bu adım eksikse veya yanlışsa sayı değişmez. Değerin hiç güncellenip güncellenmediğini görmek için döngüyü elle izleyebilirsin. Güncellenmiyorsa sorunu bulmuşsun demektir.



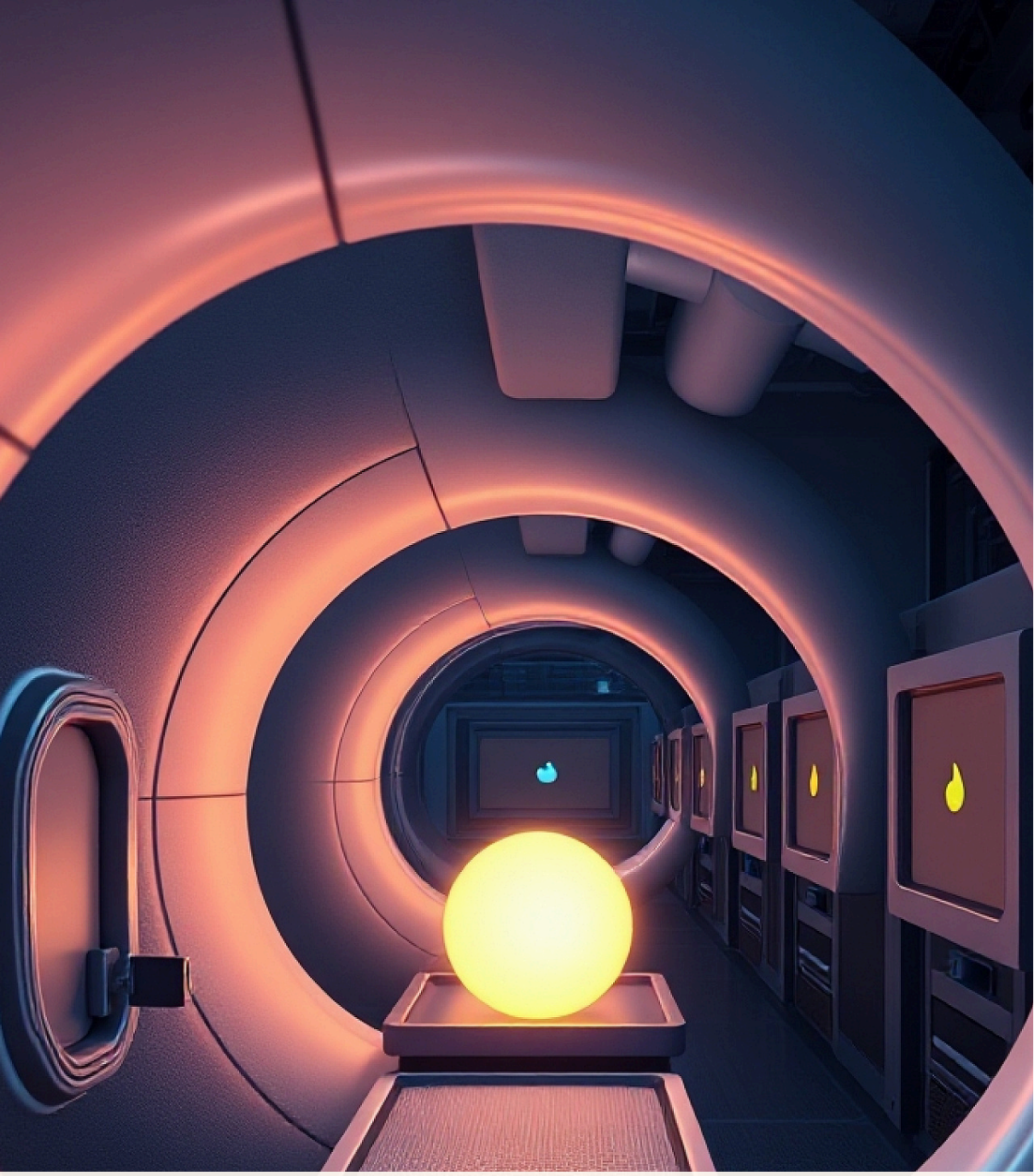
Hatayı düzeltmek için her seferinde sayıyı değiştiren adımı ekle veya düzelt. Örneğin döngünün içine sayıya bir ekleyen bir satır koy. Artık sayı artar ve sonunda sayı < 10 denetimi yanlış olur. Döngü böylece normal şekilde bitebilir. Döngünün her zaman bitiş koşuluna doğru ilerlediğinden emin ol.



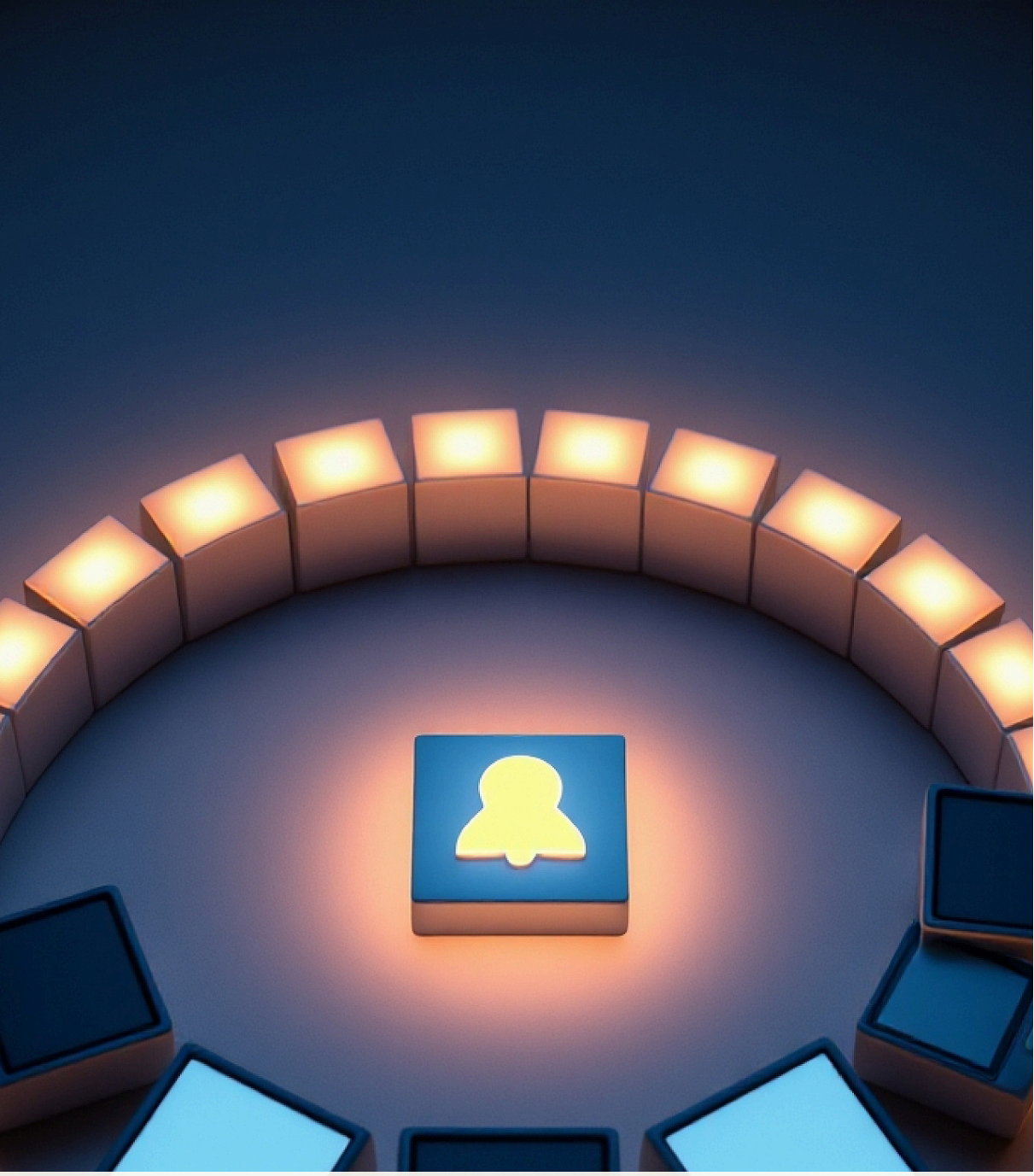
Bazen insanlar bir döngünün çalışmaya devam etmesini ister. Bu her zaman bir hata değildir. Bazı programlarda programı canlı ve duyarlı tutmak için bilinçli olarak sonsuz döngü kullanılır. Örneğin bir oyunun veya sunucunun sürekli çalışması gerekir. Bu yüzden bilinçli sonsuz döngüler yaygın ve faydalıdır.



Bir oyunun ana döngüsü, oyunu canlı tutmak için sürekli çalışır. Oyuncu girdisini denetler, oyun durumunu günceller ve ekranı çizer. Bu döngü saniyede birçok kez yinelenir. Sabit bir bitiş koşulu yoktur çünkü oyun, oyuncu çıkana kadar çalışmalıdır. Yani döngü tasarım gereği sonsuzdur.



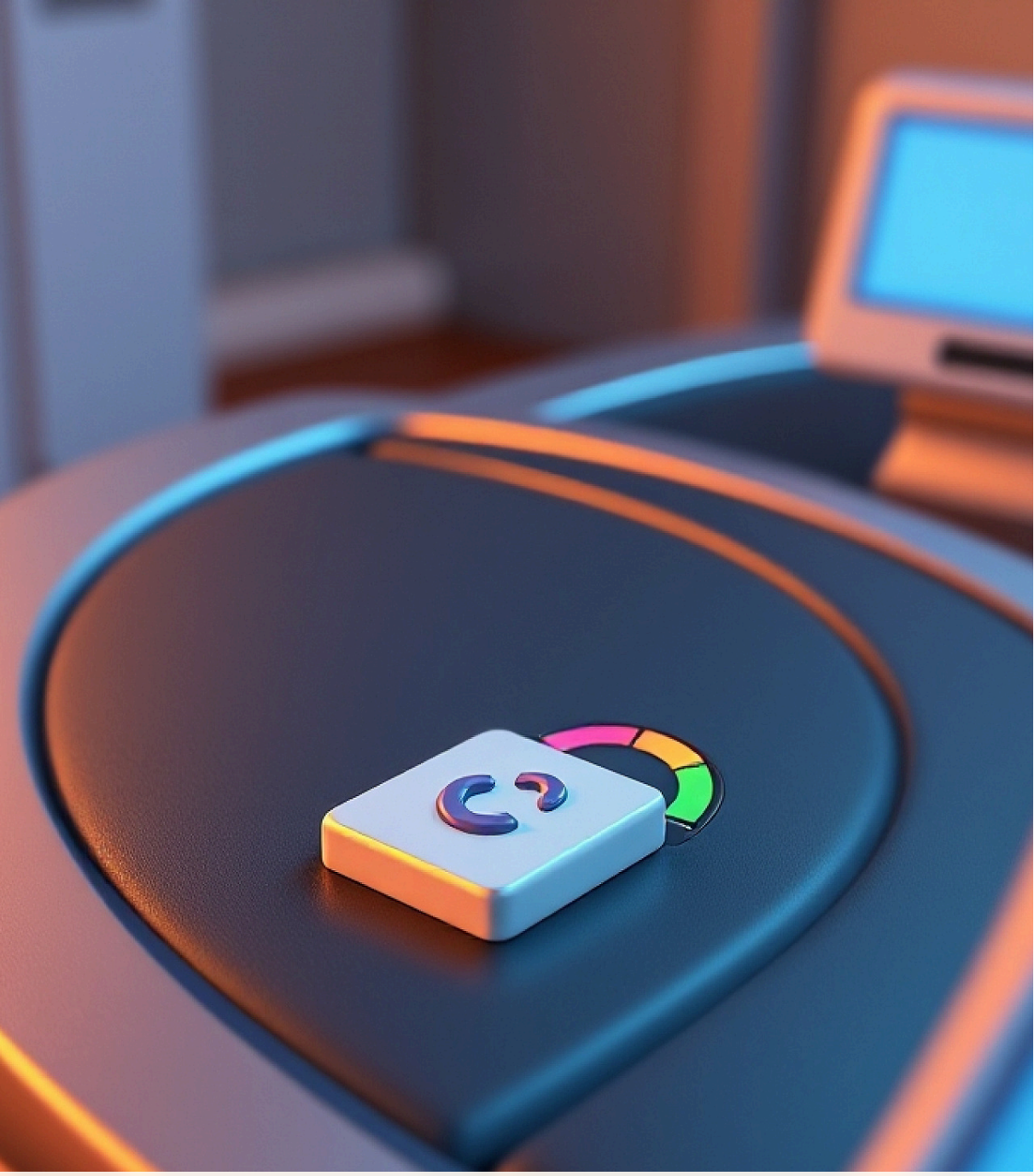
Bir sunucunun ana döngüsü, isteklere hazır olmak için sürekli çalışır. Gelen bağlantıları bekler, onları işler ve sonra beklemeye geri döner. Bu döngü, sunucu çalıştığı sürece yinelenir. Planlanmış bir sonu yoktur çünkü sunucu her zaman kullanılabilir olmalıdır. Yani bu da bir başka bilinçli sonsuz döngüdür.



Bu döngüler bir tuşa basma veya bir istek gibi olayları bekler. Bir şey olduğunda her zaman hazır olmak için yinelenirler. Tek bir olaydan sonra durmak yerine dinlemeye devam ederler. Bu, programı duyarlı kılar ve zaman içinde birçok olayı işleyebilmesini sağlar. Olayları beklemek, bilinçli bir sonsuz döngü kullanmanın önemli bir nedenidir.



Bilinçli bir döngü bile gerektiğinde durabilmelidir. Onu bitirmenin bir yolu olmalıdır; örneğin bir çıkış komutu veya bir kapatma sinyali. Örneğin bir oyun döngüsü, oyuncu çıkmayı seçtiğinde durur. Bir sunucu döngüsü, sunucu kapatıldığında durur. Durdurmanın bir yolu olmadan program sonsuza kadar çalışabilir ve kontrol edilmesi zorlaşabilir.



Çalışan bir döngü bilgisayarın zamanını ve gücünü kullanır. İyi bir döngü boş yere dönmemeli veya kaynakları boşa harcamamalıdır. Örneğin duraklamadan sürekli girdi denetleyen bir döngü çok fazla CPU kullanabilir. Daha iyi tasarımlar olayları verimli şekilde bekler veya küçük bir uyku ekler. Bu, enerji harcamadan programı duyarlı tutar.



Olağan hataları gözden geçir: denetlenen değeri değiştirmeyi unutmak ve bilinçli bir döngünün durdurulma yolu olmadan çalışmasına izin vermek. Ayrıca duraklamadan dönerek kaynakları boşa harcayan döngülerden kaçın. Hem bitiş koşulunu hem de güncelleme adımını denetle. Bilinçli döngüler için net bir çıkış olduğundan emin ol. Bu denetimler sorunlardan kaçınmana yardımcı olur.



Sonsuz döngü bitiş koşuluna asla ulaşmaz. Bir değeri güncellemeyi unutmak gibi bir hata olabilir ya da bir oyun veya sunucu döngüsü gibi bilinçli olabilir. Ama bilinçli döngüler bile durdurulabilir kalmalı ve kaynakları boşa harcamamalıdır. Güncellemeleri denetlemeyi ve durdurmanın bir yolunu sağlamayı unutma. Yanıt budur.