PROSEDÜREL MESH YARATMA ALGORİTMASI

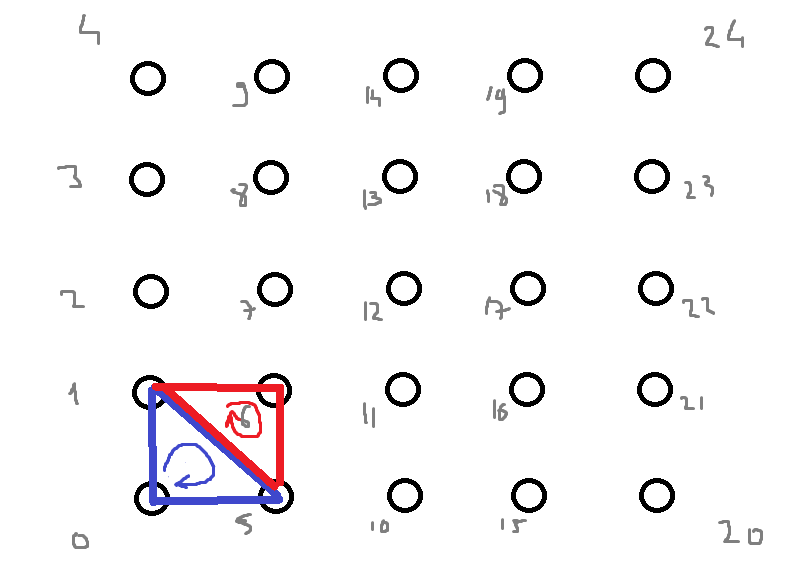
Yaratılmış olan grid alanının her Node'unu bir üçgenin vertexiymiş gibi kullanarak birim üçgenlere böldüm. (Şekil 1). Bu üçgenleri arrayde sakladım.

Bundan sonra block olarak bahsedeceğim şey bizim bütün olarak oynattığımız objedir. Blocklar boş objelerdir, sahip olacağı üçgeni blocklara assign edip o block'un child'i olur.

5 blocktan oluşan bir puzzle bölümü olcağını varsayalım. Başlangıçta bir tane block yarattım. Çözüme biraz graph ve komşuluk mantığıyla yaklaştım. Arrayin 0. Üçgenini 1. block'uma assign ettim. Daha sonra işlem sıra sıra bütün üçgenleri gezip o üçgenin komşularından bir bloğa mı gitmeli yoksa yeni bir block mu yaratılmalı diye bir random karar aldırıyorum. Levelin dinamikliği için bu kararı(%50) random yaptım. Fakat burda belirlediğimiz block sayısı kadar block yaratmamız gerektiğinden bazı kısıtlamalar yaptım. O anki assign edilmemiş üçgen sayısı eğer henüz yaratılmamış block sayısından fazlaysa üçgeni random bir komşu block'a seçip assign edecek ya da üçgen yeni bir block yaratılıp ona ait olacak. (Komşuluk; Bir üçgen eğer başka üçgenle 2 tane ortak noktaya sahipse komşudur.)

Eğer yaratılmamış block sayısı ile assign edilmemiş üçgen sayısı eşitse derhal yeni bir block yaratılıp üçgen yeni block'a assign edilecek. Ya da diyelimki random kararlar üçgenlerin yarısını assign ettikten sonra block sayısı yaratılacak block sayısına denk olmuşsa kod sıradaki üçgenleri her birinin komşu blocklara assign etme case'ine gidecektir.

Böylelikle yaratılan gridimiz her levelde dinamik olacaktır.



(Şekil 1)