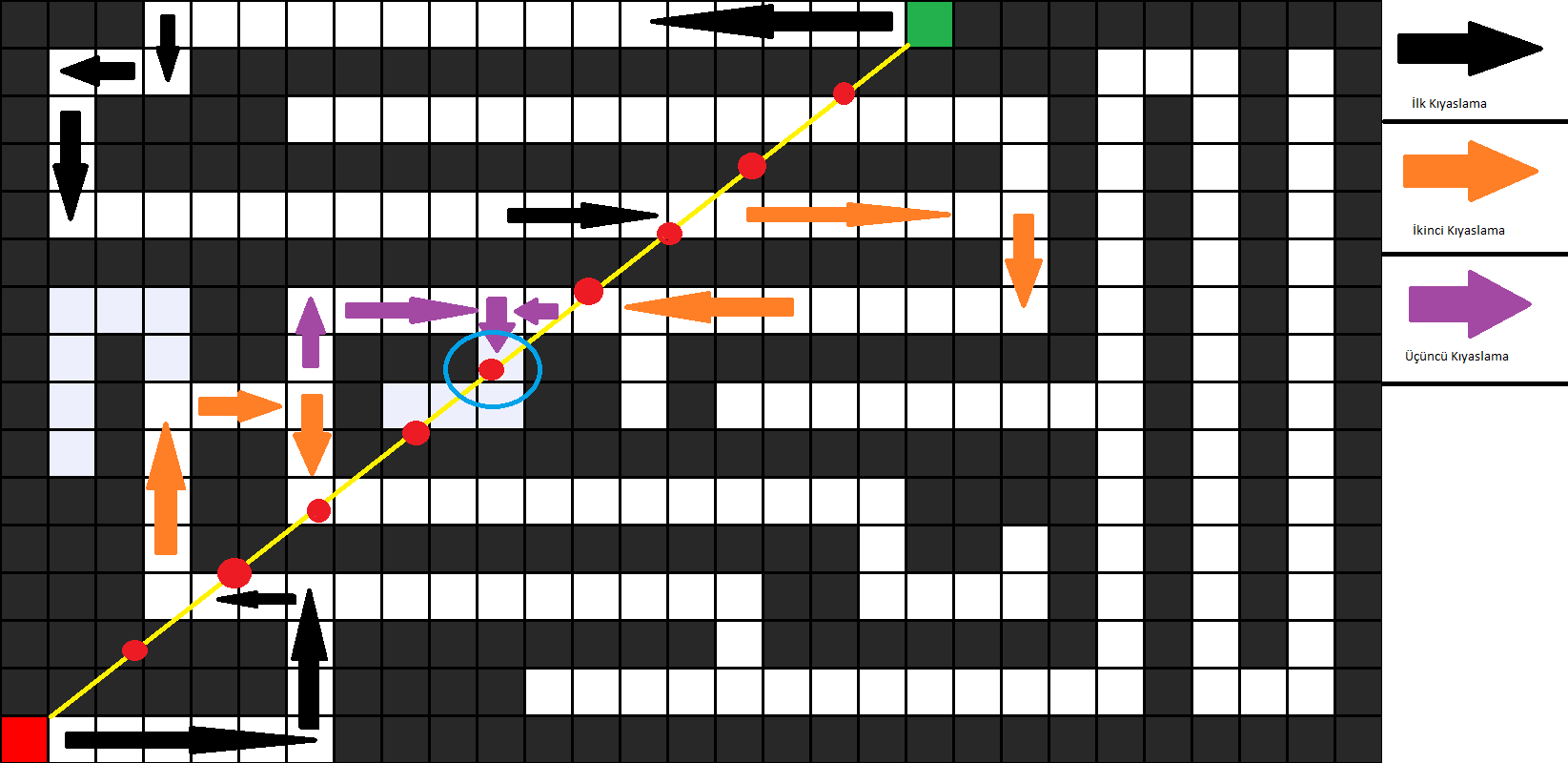
***Maze Search***

**Yunus Emre Kuşçu - 191522011**

Ben labirentte yönünü bulma konusunda maliyete değil de optimum yola ulaşmayı daha öne koydum o yüzden planlarken şöyle bir şey düşündüm. Öncelikle Breadth First Search kullanacağız. Biz başlangıç ve bitiş noktalarını labirenti sisteme okuturken zaten öğreniyoruz sadece duvarları bilmiyoruz bundan dolayı başlangıç ve bitiş noktalarını dikey olacak şekilde, arasına belirli aralıklarla (2 satır-2 sütun olabilir,3 satır-2 sütun olabilir) pointler atayacağız sonra aynı anda hem başlangıç hem bitiş noktalarından başlayıp pointten pointe ilerleyeceğiz başlangıçta ilk pointe geldiğimizde bitiş yerine dönüp oradan bitişin ilk pointine ilerleyeceğiz sonra yine bir kıyaslama yapıp yeni point noktaları oluşturup onlara göre bu süreci tekrarlayarak ilerleyeceğiz. En sonunda iki yolda bir pointte birleşecek ve yaptığımız hesaplamalardan çıkan en kısa yolu optimum olarak alıp sonuca ulaşacağız.

    Bu süreci henüz hiç test etmediğimden ne kadar verimli olur ya da ne kadar optimum sonuca erişir bilmiyorum. Ama maliyet yüksek olsa da verimliliğin yüksek olacağını düşünüyorum.



Mavi ile işaretlenmiş noktada orta yolu buluyoruz sonrasında gittiğimiz yolları kıyaslayarak birbirine en yakın uçları birleştirerek optimum sonuca ulaşabiliyoruz