**Yazılım Gereksinim Raporu**

**Proje Adı:** Labirent Mesafe Hesaplayıcı

**1. Giriş**

Bu rapor, kullanıcının labirent içindeki kendi konumu ve diğer kişilerin konumları arasındaki mesafeleri hesaplayan ve en yakın ile en uzak mesafeleri bulan bir C programının gereksinimlerini tanımlamaktadır.

**2. Amaç**

Programın amacı, kullanıcının labirent boyutlarını, kendi konumunu ve diğer kişilerin konumlarını girdikten sonra, kullanıcı tarafından seçilen mesafe hesaplama yöntemine (Öklid veya Manhattan) göre mesafeleri hesaplamak ve en yakın ile en uzak mesafeleri kullanıcıya sunmaktır.

**3. Kapsam**

Bu yazılım, aşağıdaki işlevleri yerine getirmelidir:

Labirentin boyutlarını (satır ve sütun sayısı) kullanıcıdan almalıdır.

Labirentteki nokta sayısını (diğer kişilerin sayısı) kullanıcıdan almalıdır.

Kullanıcının kendi konumunu (satır ve sütun) almalıdır.

Diğer kişilerin konumlarını (satır ve sütun) almalıdır.

Kullanıcının mesafe hesaplama yöntemini seçmesini sağlamalıdır (Öklid veya Manhattan).

Her bir kişi ile kullanıcı arasındaki mesafeyi hesaplamalıdır.

En yakın ve en uzak mesafeleri belirlemelidir.

Hesaplanan mesafeleri ve en yakın/uzak mesafeleri ekrana yazdırmalıdır.

Geçersiz girdi durumunda (örneğin, labirent sınırları dışında konum) hata mesajı vermelidir.

**4. Fonksiyonel Gereksinimler**

**Labirent Boyutu Girişi:**

Kullanıcı, labirentin satır ve sütun sayısını girebilmelidir.

Girdi değerleri pozitif tamsayı olmalıdır.

**Nokta Sayısı Girişi:**

Kullanıcı, labirentteki nokta sayısını girebilmelidir.

Girdi değeri pozitif tamsayı olmalıdır.

**Kullanıcı Konumu Girişi:**

Kullanıcı, kendi konumunu (satır ve sütun) girebilmelidir.

Girdi değerleri, labirent boyutları içinde olmalıdır.

**Diğer Kişilerin Konumu Girişi:**

Kullanıcı, diğer kişilerin konumlarını (satır ve sütun) girebilmelidir.

Girdi değerleri, labirent boyutları içinde olmalıdır.

**Mesafe Hesaplama Yöntemi Seçimi:**

Kullanıcı, Öklid veya Manhattan mesafe hesaplama yöntemini seçebilmelidir.

**Mesafe Hesaplama:**

Program, seçilen yönteme göre her bir kişi ile kullanıcı arasındaki mesafeyi hesaplamalıdır.

Öklid mesafesi için sqrt((x1​−x2​)2+(y1​−y2​)2) formülü kullanılmalıdır.

Manhattan mesafesi için ∣x1​−x2​∣+∣y1​−y2​∣ formülü kullanılmalıdır.

**En Yakın ve En Uzak Mesafe Belirleme:**

Program, hesaplanan mesafeler arasından en yakın ve en uzak olanları belirlemelidir.

**Sonuçların Gösterimi:**

Program, her bir kişinin mesafesini ve en yakın/uzak mesafeleri ekrana yazdırmalıdır.

**Hata Yönetimi:**

Program, geçersiz girdi durumunda (örneğin, labirent sınırları dışında konum) hata mesajı vermelidir.

**5. Fonksiyonel Olmayan Gereksinimler**

**Kullanılabilirlik:**

Program, kullanıcı dostu bir arayüze sahip olmalıdır.

**Performans:**

Program, mesafeleri hızlı ve verimli bir şekilde hesaplamalıdır.

**Güvenilirlik:**

Program, doğru sonuçlar üretmelidir.

**Taşınabilirlik:**

Program, farklı işletim sistemlerinde derlenip çalıştırılabilir olmalıdır.

**6. Girdi/Çıktı Gereksinimleri**

**Girdi:**

Labirent boyutları (satır, sütun).

Nokta sayısı.

Kullanıcı konumu (satır, sütun).

Diğer kişilerin konumları (satır, sütun).

Mesafe hesaplama yöntemi seçimi.

**Çıktı:**

Her bir kişi ile kullanıcı arasındaki mesafe.

En yakın mesafe.

En uzak mesafe.

Hata mesajları.

**7. Varsayımlar ve Kısıtlamalar**

Kullanıcı, geçerli tamsayı değerleri girecektir.

Labirent boyutları ve nokta sayısı pozitif tamsayı olacaktır.

Program C dilinde yazılacaktır.