**T.C.**

**ESKİŞEHİR OSMANGAZİ ÜNİVERSİTESİ**

**MÜHENDİSLİK-MİMARLIK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**



**NESNE TABANLI PROGRAMLAMA I PROJE RAPORU**

GEZGİN ROBOT DENETİM SİSTEMİ

AŞAMA ||

152120181063 GAMZE BİLMEZ

152120181070 MELİH BUCAK

152120181040 MELİSA ÖZEL

152120171103 EŞREF BURAK KAYA

**OCAK 2021**

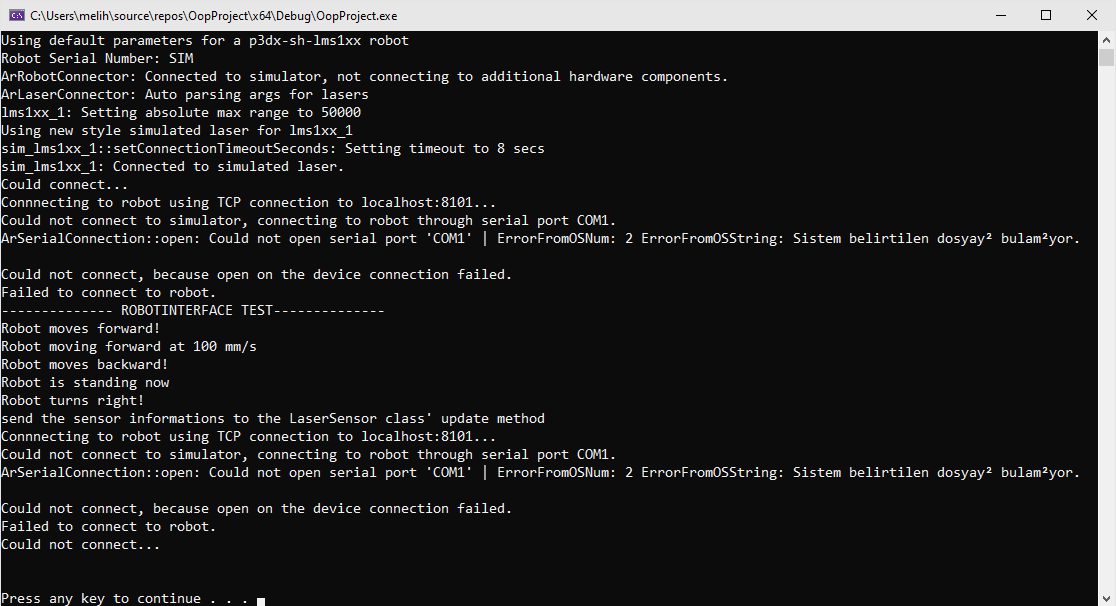
**1.GİRİŞ**

Projemizin bu aşamasında abstract sınıf yapısını kullandık ve bu sayede farklı bir robot eklemek istediğimizde, aynı robota ait farklı bir API kullanmak istediğimizde veya yeni bir sensör eklemek istediğimizde miras ile yeni sınıflar oluşturabilecek ve diğer yazılımlarımız bundan etkilenmeyecektir.

**2.TASARIM**

CLASS RangeSensor: RangeSensor abstract sınıfı, laser ve sonar olmak üzere her iki sensöre aynı arayüz ile ulaşılabileceği gibi, yeni eklenecek sensörler için de yazılımın etkilenmemesini sağlıyor. Ayrıca RangeSensor sınıfı üzerinden Sensör verilerinin robottan gelen veriler ile güncellenmesi sağlanabiliyor.  
  
CLASS RobotInterface: RobotInterface abstract bir sınıftır. Farklı bir robot ait bir API ya da aynı robota ait farklı bir API kullanıldığında, RobotInterface sınıfından miras alan başka bir sınıf eklenecektir. Bu eklentiden, yazılımımızın geri kalanı etkilenmeyecektir.

CLASS PioneerRobotInterface: PioneerRobotAPI ile doğrudan ilişkilidir ve RobotInterface abstract sınıfını miras alır.

  
CLASS RobotControl: Sınıfı robotla ilgili işlemleri RobotInterface üzerinden işletecektir. Bunun yanında yeni fonksiyonlarda eklenmiştir.

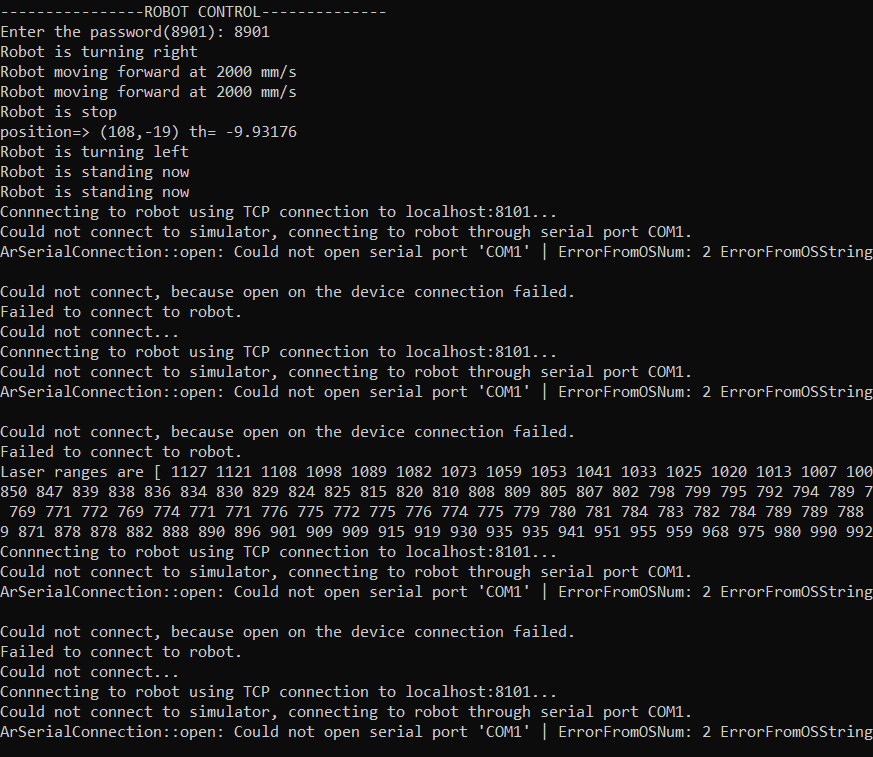
addToPath() fonksiyonu, robotun bulunduğu konumun path’e eklenmesini sağlar.

clearPath() fonksiyonu path’deki konumları temizlemektedir.

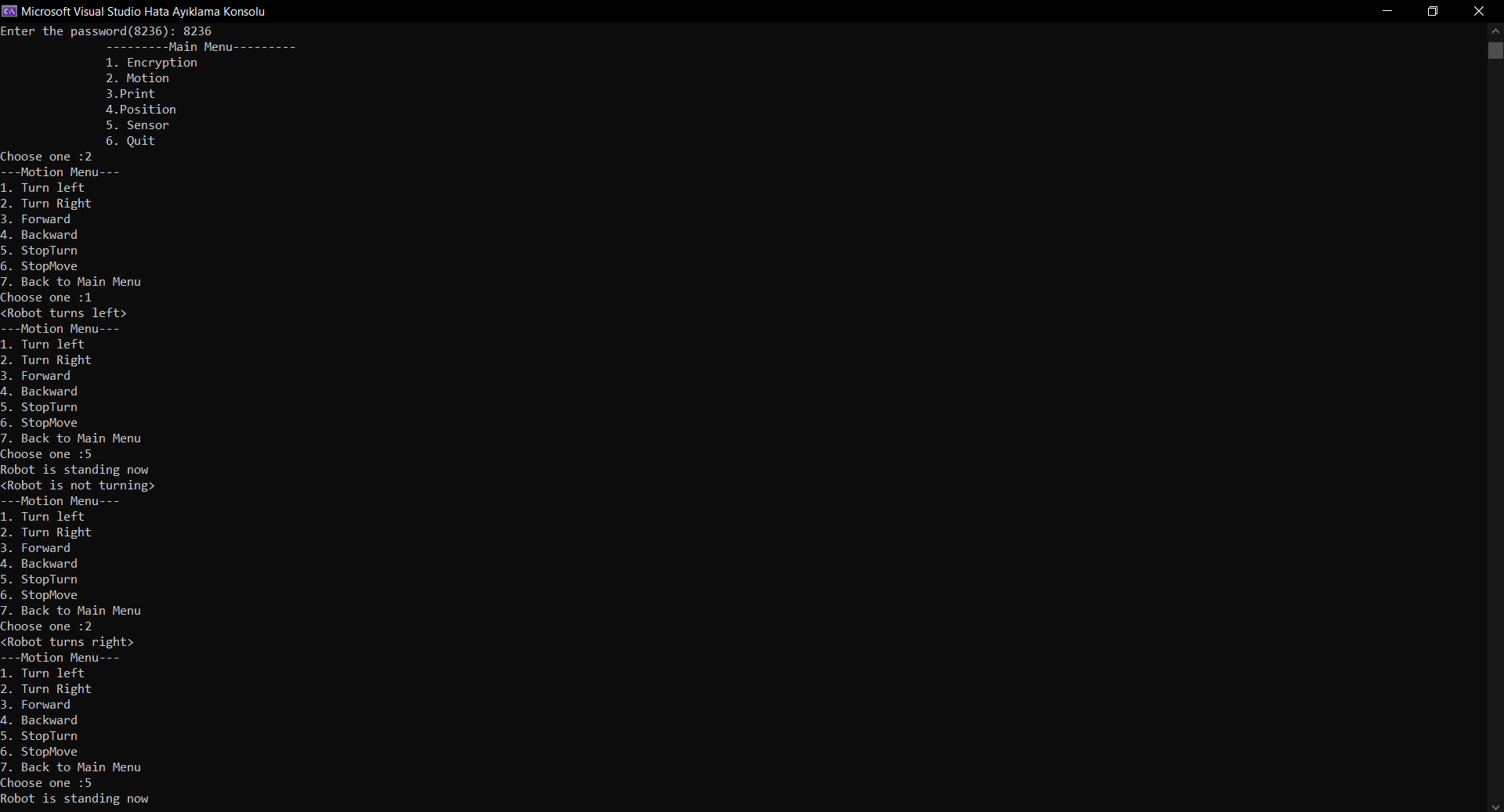
recordPath() fonksiyonu Path nesnesinde yüklenmiş olan konumları dosyaya yazdırmaktadır.

openAccess() fonksiyonu, RobotControl’e erişim için kullanılmaktadır. Doğru şifre giriş yapılmadığı takdirde, RobotControl sınıfının işlem yapmamasını sağlar. Doğru şifre verildi ise, tüm üye fonksiyonlar yapması gereken işlemi yapacaktır.

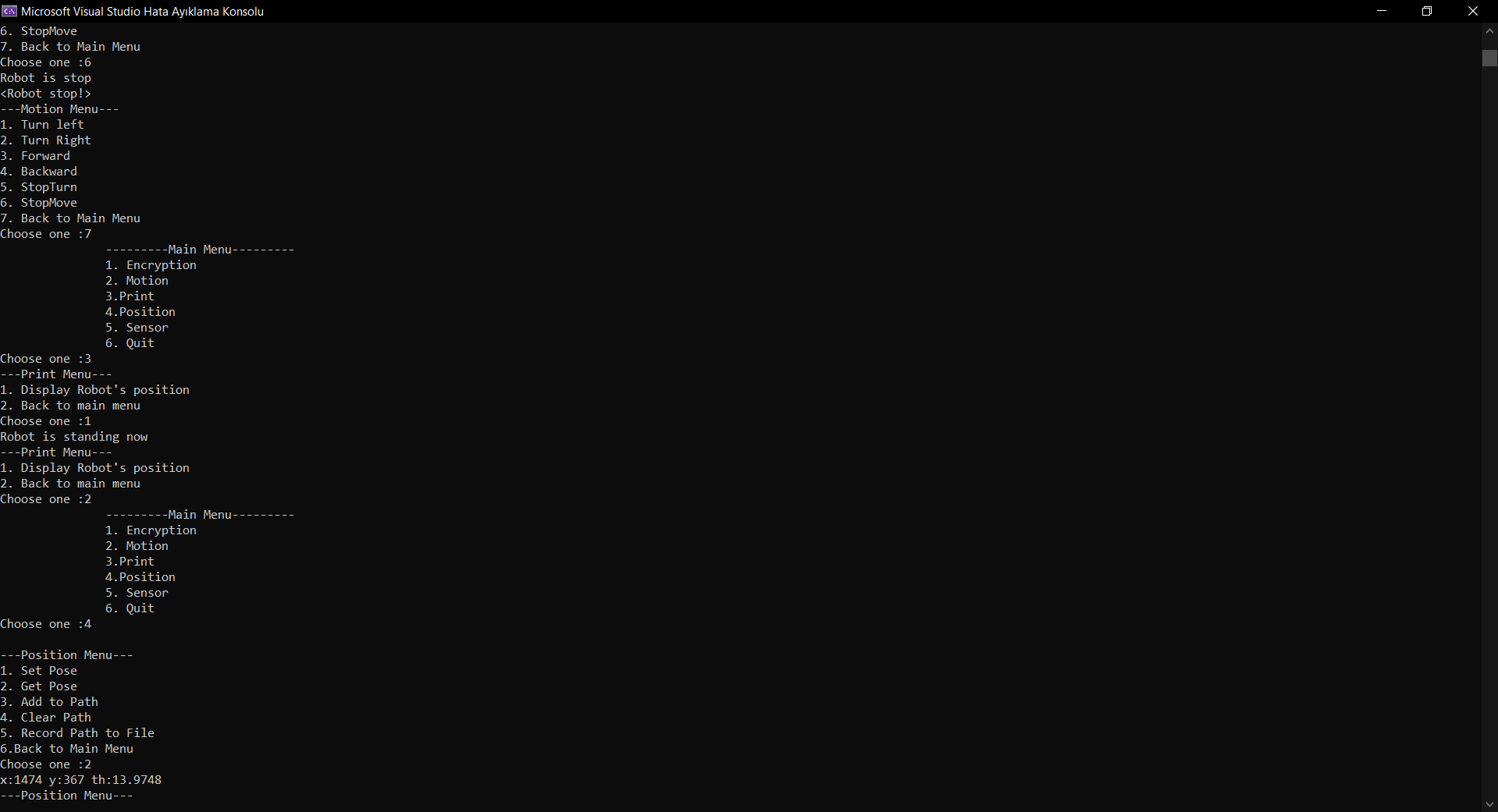
closeAccess() fonksiyonu ile doğru şifre verildiğinde, erişim kapatılacaktır.

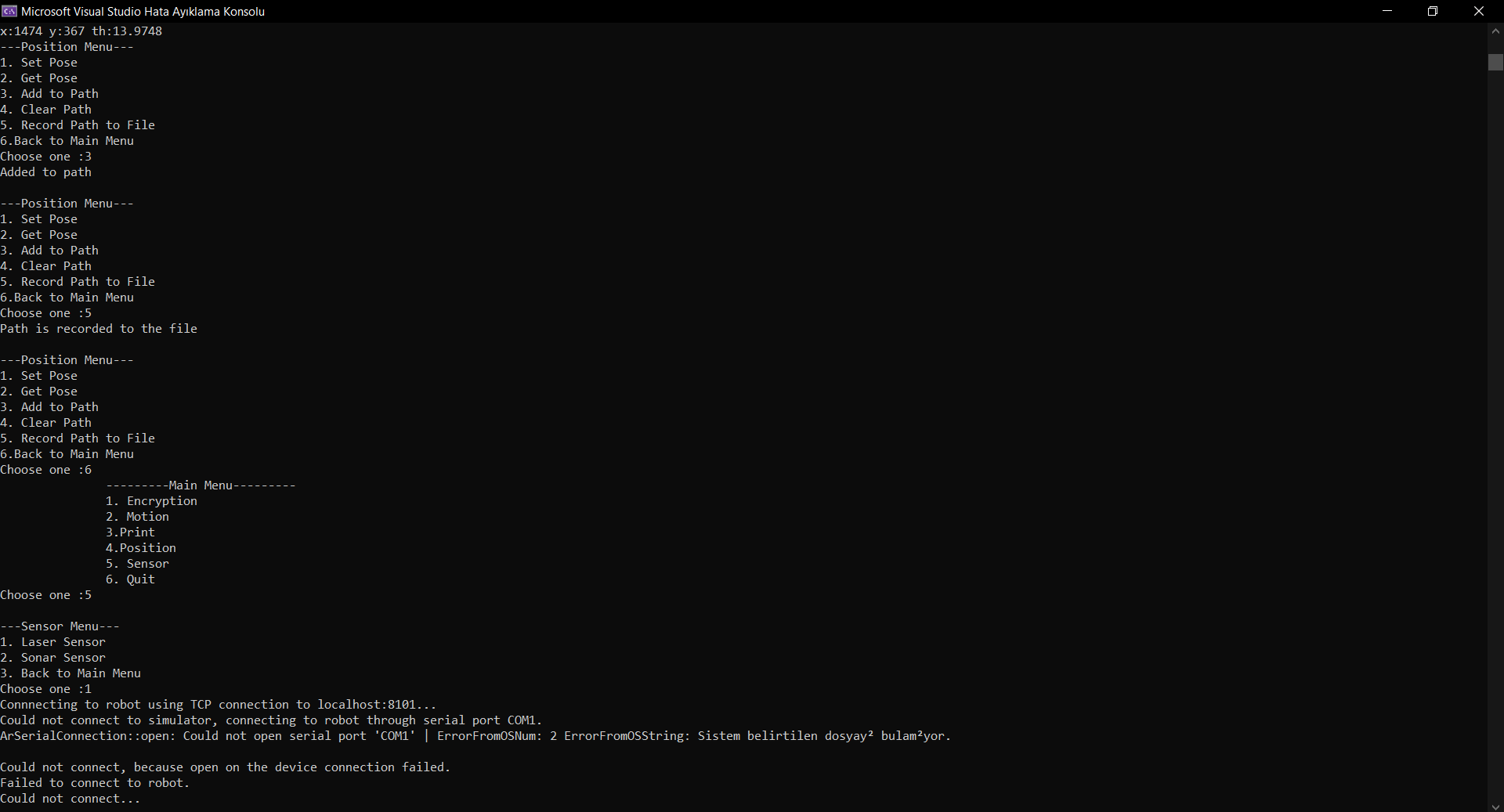


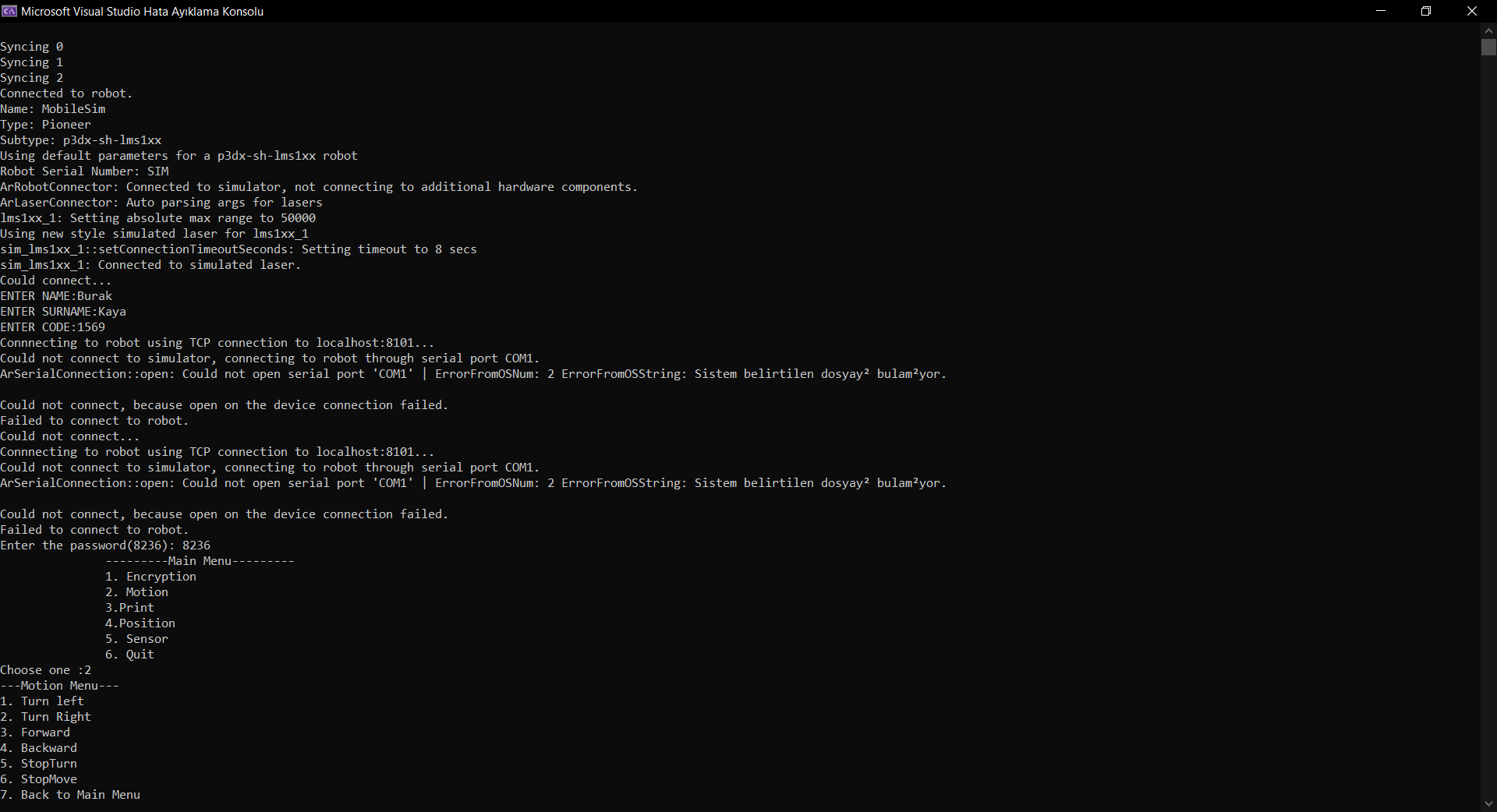
**Menu Class**

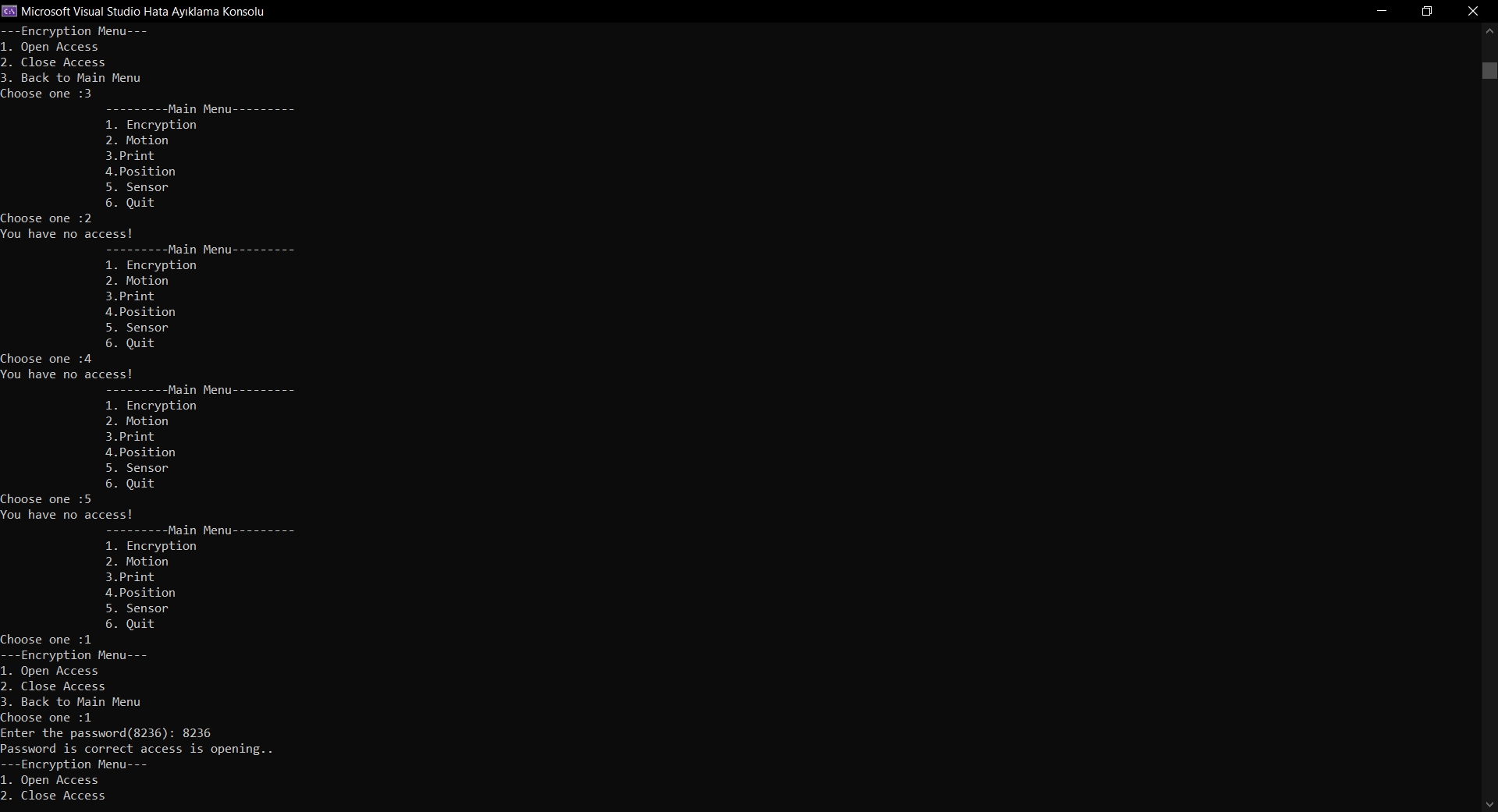


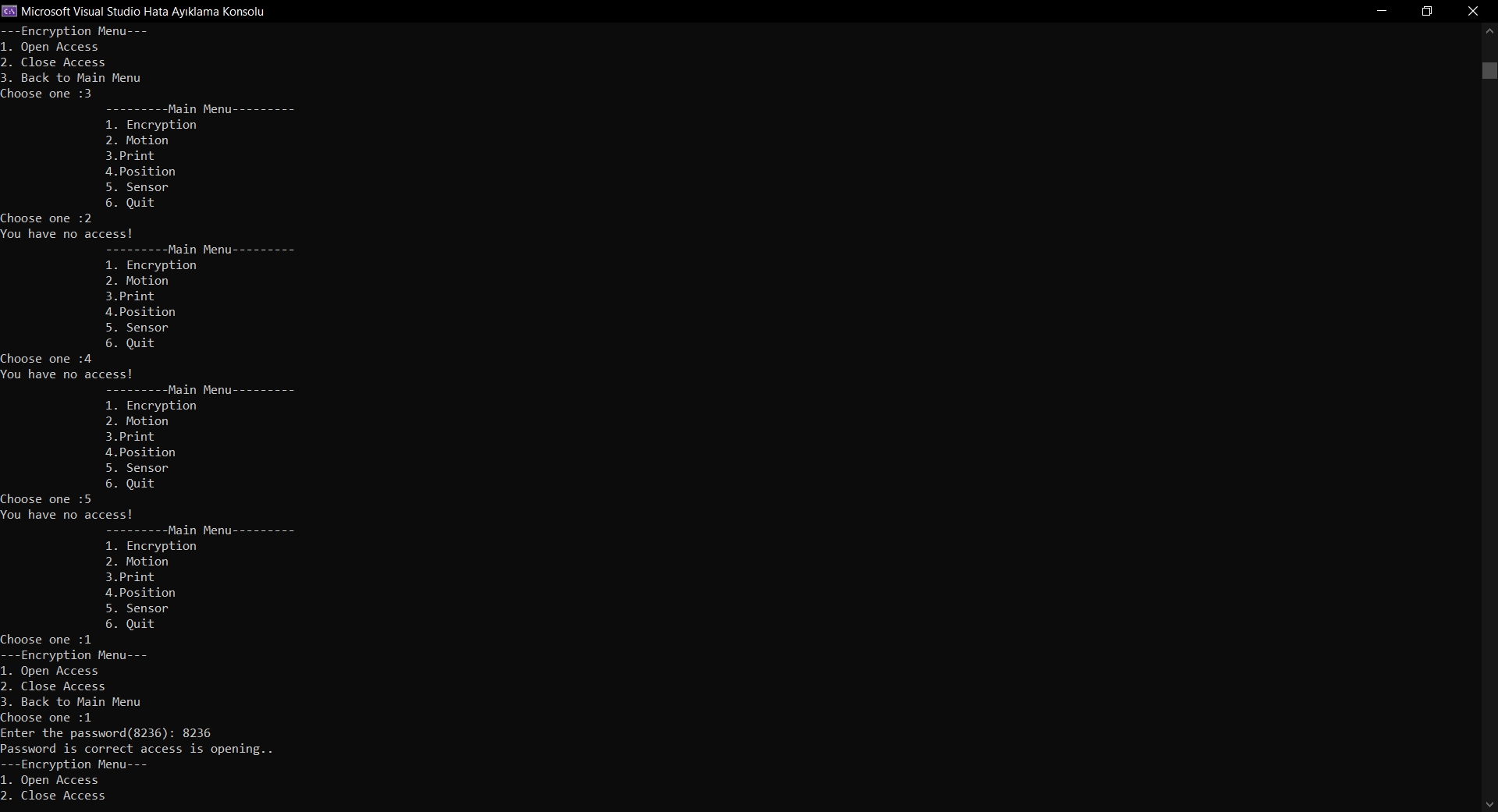


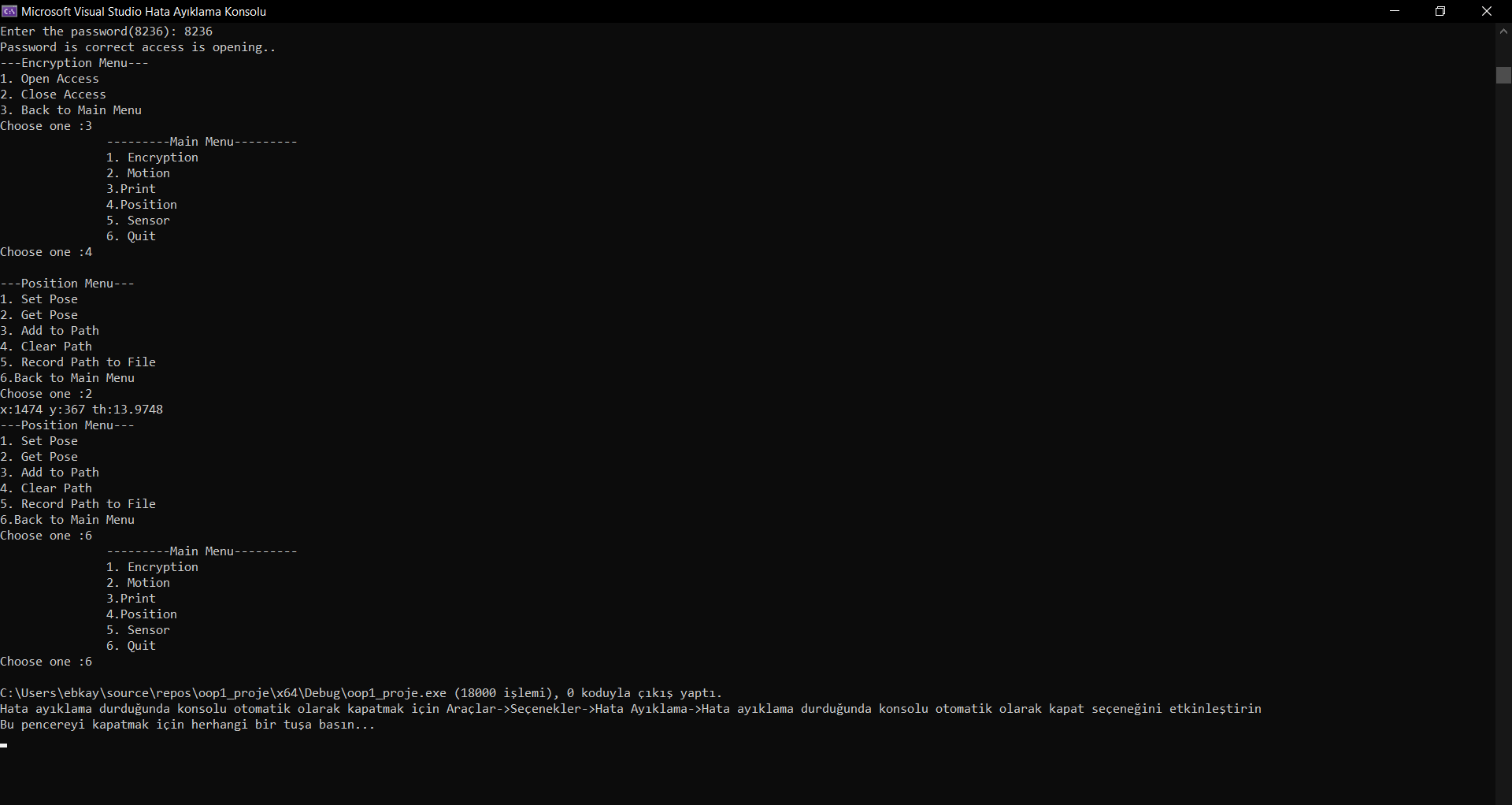




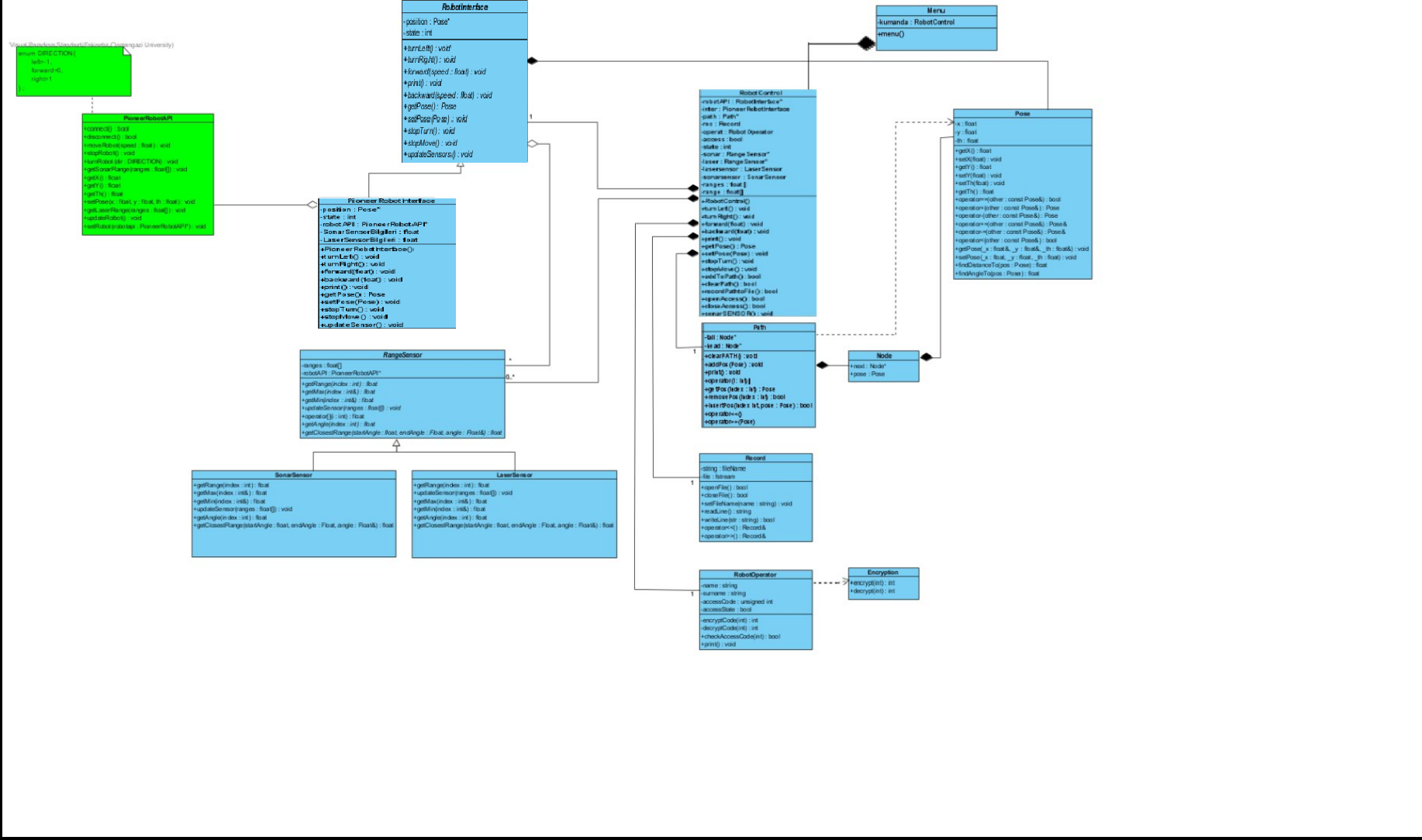








**UML**



|  |  |
| --- | --- |
| TAKIM ÜYELERİ | GÖREVLERİ |
| EŞREF BURAK KAYA | RobotControl class ,Menu class, Record class |
| MELİSA ÖZEL | RangeSensor-SonarSensor- LaserSensor class |
| GAMZE BİLMEZ | RobotControl class, Menu class,Doxygen |
| MELİH BUCAK | RobotInterface-PionerRobotInterface class,Menu class,UML |

**3. SONUÇLAR VE DEĞERLENDİRMELER**

Grup üyeleri kendine ait class’ları ve test programlarındaki güncelleştirmeleri yapıp, yeni eklenen abstract sınıflar için uygun kodları yazmıştır.

Grup olarak projemizin bu aşamasında abstract sınıfını nasıl kullanabileceğimiz, polimorfizm yapısının getirdiği avantajlar ve dezavantajlar konularındaki eksikliklerimizi giderdik ve konuları daha iyi kavradık.

Takım çalışmasında avantajların yanı sıra dezavantajların da olduğu görülmüştür. Polimorfizm yapısına uygun olarak kodumuzu güncellerken, herkesin farklı bir class kodunu yazmasından kaynaklı olarak proje sürecinin uzaması ve her iki sensör işlemlerinde de sorunlar yaşandı.  
Sonuç olarak grup olarak çalışma becerimiz bu projeyle daha da arttı.