**7.1    Prototípus interface-definíciója**

A prototípus a szkeletonhoz hasonlóan konzolos felületen keresztül lesz irányítható, valamint konzolra írja ki az adatokat. Az irányítás az általunk meghatározott különböző parancsokkal lesz lehetséges. A prototípus már nem csak függvényhívás-sorrendet fog mutatni (szekvenciákat), hanem a játék tényleges működését. A teszteléshez teszteseteket hozunk létre, melyek konkrét parancssorozatból állnak, így automatikusan tudják a játék menetét befolyásolni. Minden teszteset kimenetének egyeznie kell az általunk előre meghatározott referencia kimenettel. Ezen referencia kimenetek majd a prototípus részletes tervében kerülnek bemutatásra. Fontos, hogy a tesztek mindenki számára egyértelmű eseménysorozatot tartalmazzanak, valamint többször, azonos formában megismételhetőek legyenek, az utóbbi érdekében a véletlen paraméterek is beállíthatóak (a véletlenszerűség továbbá kikapcsolható). A teszteseteket fájlból is be lehet olvasni. Minden csapattag ezen elveket szem előtt tartva látott a tervezéshez.

7.1.1    Az interfész általános leírása ??

7.1.2    Bemeneti nyelv

loadCommands

Leírás: Beolvassa és végrehajtja a *commands.txt* fájlban található parancsokat

Opciók: commands.txt

loadMap

Leírás: Beolvassa a *map.txt* fájból az aktuális tesztpályát

Opciók: map.txt

moveCivil

Leírás: Az *n.* CivilCar a következő kereszteződésnél a lehetőségek közül az *i.* cellát választja

Opciók: n, i

movePolice

Leírás: Az *n.* Policeman a következő kereszteződésnél a lehetőségek közül az *i.* cellát választja

Opciók: n,i

moveRobber

Leírás: A rabló a következő kereszteződésnél a lehetőségek közül az i. cellát választja

Opciók: n,i

turnOffSpeed

Leírás: Autók sebességének kikapcsolása/bekapcsolása (sw paraméterrel). (Minden autó minden tick-re lép, vagy sebességüknek megfelelően).

Opciók: sw

setSign

Leírás: Beállítja az *n.* lámpa/ állapotát *blocking* állapotra, és beállítja a megfelelő számlálókat.

Opciók: n, blocking

tick

Leírás: Az egész játékteret *n*-el lépteti (*n* darab tick)

Opciók: n

exit

Leírás: Kilép a tesztelésből.

Opciók: n

writeMap

Leírás: Kiírja a pálya állását. (Bementként is értelmezhető formátumban) (Lásd pálya szerkezete)

Opciók: -

7.1.3    Kimeneti nyelv

Hibaüzenetek:

**x Error : FileNotFound:** Nem található a megadott bemeneti fájl.

**x Error : Unknown Command:** Nem értelmezhető parancs.

**x Error : Mismatching Parameters:** Hibás paraméterek a parancsnak.

**x Error : Invalid Map Structure:** Hibás pálya leírás.

**x Error : Invalid Event Occured:** Hibás esemény egy játékelem által.

Események:  
- A writeMap parancsra kiíródik a pálya később ismertetett formátumban.   
- Bármely játékelem, ha cselekvést hajt végre, kiír egy üzenetet a konzolra:

o Event : ElemTípus ElemNév [Elemsorszám/-](ha van) Esemény

(pl: **o Event:** Robber r Went to 1. Cell. **o Event:** PoliceMan p1 0 Stepped.)

A pálya formátuma: A pályán az alábbi objektumok lehetnek

CityEntry - E[#]

CityExit - X[#]

InterSection - I[#]

Bank - B

HidingPlace - H

RoadCell - F

StopSign - S[#]

TrafficLight T[#]

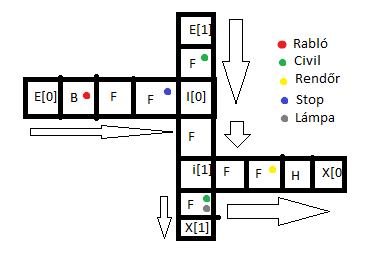
CivilCar - C[#]

PoliceMan - P[#]

Robber - R

A név után látszik, hogy hogyan is jelöljük a pálya fájljában az egyes elemeket, valamint néhány után van [#], amivel jelezzük, hogy ezeket az elemeket számokkal kell ellátnunk, hogy a későbbiek során vezérelni tudjuk őket, illetve, hogy a pályaépítés során azonosíthatók legyenek.

A pálya reprezentációban minden sorban egy-egy út szerepel, a két szélén az őt körbevevő Intersection elemekkel. Ha egy útelem (Cell) tartalmaz egy objektumot (Vehicle vagy Sign típusút), akkor kapcsos zárójelek között kell rögzíteni mögé (ha többet is, akkor vesszővel elválasztva kapcsos zárójelek között. Ilyen módon egyértelműen generálható a pálya az általunk megadott leírásból. A tábla és lámpa elemek ugyanazon sorszámozáson osztoznak, hiszen vezérlés szempontjából nem különböznek. A menetirány balról jobbra halad minden út esetében.



Példa:

E[0]B{R}

B{R}FF{S[0]}I[0]

E[1]F{C[0]}I[0]

I[0]FI[1]

I[1]FF{P[0]}H

HX[0]

I[1]F{C[1],T[1]}X[1]

Akkor jó egy pálya, ha:

- Minden olyan intersection, amiből csak kifelé indul út, az CityEntry.

- Tinden olyan intersection, amibe csak befelé megy út, az CityExit.

- Tábla vagy lámpa csak az út utolsó celláján lehet.

7.1    Tesztelési terv

A programot az összes tesztesetre lefuttatjuk, és a kimenetet az általunk írt program segítségével összehasonlítjuk a referencia kimenettel.

|  |  |
| --- | --- |
| **Teszt-eset neve** | Car vs Signs |
| **Rövid leírás** | Egy autó odaér egy kereszteződés elé, valamint ott egy stoptábla megállítja. |
| **Teszt célja** | A táblák és lámpák működésének tesztelése |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Car vs Car |
| **Rövid leírás** | Egy autó utolér egy másikat, majd lelassít mögötte. |
| **Teszt célja** | Sebességkülönbségek kezelésének vizsgálata, találkozás. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Robber vs Car |
| **Rövid leírás** | A rabló belefut egy civilbe. |
| **Teszt célja** | A rabló ütközésének tesztelése, játék vége. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Car Chooses Direction |
| **Rövid leírás** | Egy autó választ a lehetőségek közül, hogy merre haladjon egy elágazás közepén. |
| **Teszt célja** | Kanyarodás tesztelése. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Robber Chooses Direction |
| **Rövid leírás** | A rabló kiválasztja, hogy merre haladjon egy elágazásnál. A többi autóéhoz képest máshogy működik. |
| **Teszt célja** | Rabló kanyarodásának tesztelése. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Policeman vs Robber |
| **Rövid leírás** | A rendőr abba az utcába érkezik, ahol a rabló tartózkodik. |
| **Teszt célja** | Rabló letartóztatásának tesztelése. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Robber Gets from Bank to HidingPlace |
| **Rövid leírás** | A rabló eljut a banktól a rejtekhelyre, közben a többi jármű megadott módon viselkedik. |
| **Teszt célja** | Játék felépítésének tesztelése, rabló és egyéb járművek mozgásának tesztelése, sebességek, lámpák tesztelése, HidingPlace elérésénél játék leállításának tesztelése. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Car vs CityExit then Regenerate |
| **Rövid leírás** | Egy autó kilép a városból egy CityExit-en, majd egy CityEntry-n visszatér. |
| **Teszt célja** | CityExit és CityEntry, valamint újragenerálás tesztelése. |

7.2    Tesztelést támogató segéd- és fordítóprogramok specifikálása

A tesztelés közben a pálya bármikor kiíratható a megadott formátumban, lementhetjük az aktuális játékállást egy későbbi teszthez. Mivel össze kell hasonlítanunk a futások eredményét a referencia-kimenettel, ezért írunk egy programot, mely el tudja végezni az összehasonlítást két szöveges fájl között.