U ovom dokumentu se nalazi spisak dostupnih registara i instrukcija, sintaksnih pravila i specifikacije programa.

* *Dostupni registri:*

**RAX, RCX, RSI, RDI**

* *Instrukcije:*
  + *instrukcije sa 2 argumenta:*

**MOV, ADD, SUB, AND, OR, CMP**

* + *instrukcije sa 1 argumentom*:

**JE, JNE, JL, JGE, NOT**

* + *instrukcije bez argumenata:*

**LOAD, STORE**

Argumenti instrukcija sa 2 argumenta (**MOV, ADD, SUB, AND, OR, CMP**) mogu biti sljedece kombinacije:

* + *registar, registar*
  + *registar, memorijska adresa*
  + *memorijska adresa, registar*
  + *registar, konstanta*
  + *memorijska adresa, konstanta*
  + *dodatno sa instrukciju CMP osim ovih kombinacija vaze dvije dodatne:*
    - *konstanta, registar*
    - *konstanta, memorijska adresa*

Argument instrukcija **JE, JNE, JL, JGE, NOT** moze biti samo labela, dok za instrukciju **NOT** moze biti samo registar.

Instrukcije **LOAD** i **STORE** se zadaju bez argumenata.

Memorija je organizovana sekvencijalno, inicijalno se dodijeli 10 memorijskih lokacija (0x0, …, 0x9), a zatim se moze vrsiti prosirenje memorije ukoliko se u programu nalazi instrukcija koja sadrzi memorijsku adresu koja nije sadrzana u kolekciji (DictionaryMemory).

Svi registri i memorijske lokacije su inicijalno postavljeni na 0.

Skripta je validna ako se prije svih instrukcija nalazi kljucna rijeci ***start***, u suprotnom nece se moci izvrsiti skripta. Sva ostala sintaksna pravila su neposredno vezana za instrukcije i labele, instrukcije su validne ako je procitana odgovarajuca instrukcija podrzana programom, i ako joj je proslijedjen odgovarajuci broj argumenata, dok je labela validna ako se zavrsava sa **: (dvotackom**).

Takodje, ako se u skripti procita komanda **IN**, ta komanda predstavlja sistemski poziv. Trazice se od korisnika da sa konzole unese odgovarajucu instrukciju. Taj unos ce biti moguc sve dok korisnik ne pritisne ‘***enter (‘\n’)’*** , a zatim ce se provjeriti sintaksa linije i ukoliko je ispravna izvrsice se. Sistemskim pozivom je omoguceno da se izvrsi maksimalno jedna instrukcija, a minimalno nijedna.

Izvrsna aplikacija se nalazi na : AssemblerSimulation\AssemblerSimulation\bin\Debug\netcoreapp3.1 pod nazivom “AssemblerSimulation.exe”.

Takodje, na ovoj putanji se nalazi par skripti namijenjenih za testiranje programa.