

Također, ukoliko se na nekoj lokaciji ne nalazi nikakva instrukcija, nego samo neki običan broj, prve tri cifre tog broja prosto ignorišemo (one su namijenjene samo za kod instrukcije), tako da ako je sadržaj memorijske lokacije 01000000000000000000000001100 i potrebno je pročitati broj s te lokacije, smatramo da ćemo pročitati broj 12.

Vaš zadatak je da izvršite mašinski program. Preciznije, bit će vam poznato koliko imate memorije na početku (tj. koliko vam je memorijskih lokacija dostupno) i znat ćete stanje memorije, tj. koji se broj nalazi na svakoj lokaciji. Vi trebate da izvršite program, za koji ćemo pretpostaviti da počinje od prve memorijske lokacije. Program staje onda kada naiđe na instrukciju 11111111111111111111111111111111. Vaš zadatak je da ispišete stanje memorije nakon izvršenog programa.

Uzmimo, na primjer, da imamo memoriju koja se sastoji od 10 memorijskih lokacija (podsjetimo, svaka lokacija je jedan 32-bitni broj). Neka je stanje memorije sljedeće:

```
00000000000000000000000000000000110
01000000000000000000000000000000111
001000000000000000000000000000001000
000000000000000000000000000000001001
011000000000000000000000000000001010
11100000000000000000000000000000101
01000000000000000000000000000000110
000000000000000000000000000000000000
10000000000000000000000000000000010
111111111111111111111111111111111111
```

Prva instrukcija je LOAD 6, (prva 3 bita su 000, što je kod za instrukciju LOAD, dok je parametar binarno zapisan broj 6). Nakon te instrukcije u pomoćnom registru imamo broj 5, jer se on nalazi na memorijskoj lokaciji 6. Druga instrukcija je ADD 7, pa ćemo broj iz pomoćnog registra (a to je 5), sabrati sa brojem 6, koji se nalazi na memorijskoj lokaciji 7. Rezultat je 11 i sada se u pomoćnom registru nalazi broj 11. Sljedeća naredba je STORE 8, pa ćemo broj 11 smjestiti na memorijsku lokaciju 8. Nakon toga slijedi naredba LOAD 9. Na memorijskoj lokaciji 9 se nalazi broj -2 (negativan je broj jer mu je prva cifra 1), pa se on kopira u pomoćni registar. Dolazimo do naredbe BNEG 10. Pošto je -2, broj u pomoćnom registru, negativan, izvršavanje nastavljamo od memorijske lokacije 10. Na memorijskoj lokaciji 10 su sve jedinice, pa program staje. U toku programa smo izmijenili memorijsku lokaciju 8, na nju smo upisali broj 11, pa stanje memorije sada izgleda:

```
00000000000000000000000000000000110
01000000000000000000000000000000111
001000000000000000000000000000001000
000000000000000000000000000000001001
011000000000000000000000000000001010
11100000000000000000000000000000101
01000000000000000000000000000000110
000000000000000000000000000000001011
10000000000000000000000000000000010
111111111111111111111111111111111111
```

Dakle, promijenjen je samo broj na memorijskoj lokaciji 8.

[illegible]