

```
POP BC ;shrani vrednost s sklada začasno v BC
PUSH HL ;porini vrednost HL na sklad
PUSH BC ;spravi prejšnjo vrednost...
POP HL ;...in jo poberi v HL.
```

Isto dosežemo z enim samim ukazom

```
EX (SP), HL ,
```

kar je precej bolj enostaven način.

```
UPORABA: Vrednosti v registrih B in C bi želeli zamenjati med
seboj. Najhitrejšo rešitev nam ponuja uporaba sklada (natan-
čneje: ukazov PUSH, POP ter INC SP - "povečaj kazalec sklada").
Poskusite jo najti sami, preden pogledate, kaj smo napisali!
C5 PUSH BC ;na sklad vedno porine najprej visoki in
nato nizki zlog
C5 PUSH BC ;SP kaže zdaj naslov vrednosti C na skladu
33 INC SP ;SP kaže zdaj naslov vrednosti B
C1 POP BC ;vedno se najprej sname nizki in nato
visoki zlog
33 INC SP ;tako uravnamo sklad
C9 RET
```

REŠITEV NALOGE O PROSTEM POMNILNIKU

Naslov, do katerega segajo program in spremenljivke v basicu, je 16-bitno število, spravljeno v E_LINE. Če torej napolnimo HL s tem naslovom, smo že na pol pota

```
LD HL, (E_LINE).
```

Zaradi prenosa moramo spustiti zastavico prenosa (najlaže z ukazom AND A, o katerem bomo govorili v naslednjem poglavju). Dobili ste petico, beri odlično, če ste pomislili na spuščanje zastavice, a niste vedeli kako:

```
AND A .
```

Zdaj odštejemo še kazalec sklada

```
SBC HL, SP.
```

Ker je naslov v kazalcu sklada višji kot naslov konca programa, bo rezultat negativen. Število moramo torej pretvoriti v pozitivno obliko. Uporabili bomo par BC (DE bi bil ravno tako dober) in vanj prestavili vrednost HL

```
PUSH HL
POP BC.
```

HL še vedno vsebuje isto vrednost (HL = BC). Da bi dobili HL = -BC, moramo BC dvakrat odšteti od HL. Ne pozabite, da je naše prvo odštevanje dvignilo zastavico prenosa in jo moramo zato znova spustiti

```
AND A
SBC HL, BC
SBC HL, BC .
```

HL zdaj vsebuje število prostih zlogov v pozitivni obliki. Da bi to število dobili kot rezultat funkcije USR, ga moramo prestaviti v BC

```
PUSH HL
POP BC
```

in se vrniti

```
RET .
```

Je bila pot do rešitve zelo težka?