

# 1. IZRADA KODA ZA IZVEDBU

Napiše se izvorni kod (source) u editoru:

```
#include <stdio.h>
int main()
{
    int a,b,c;
    scanf("%i%i", &a, &b);
    c=a+b;
    printf("%i", c);
}
```

→ spremi se  
kao program.c

Posle se kompilira: `cpp program.c program.o`. Kompiler proverio je li se dobro napisano. Kreira se objektni kod koji se sastoji od:

## TABLICA SIMBOLA

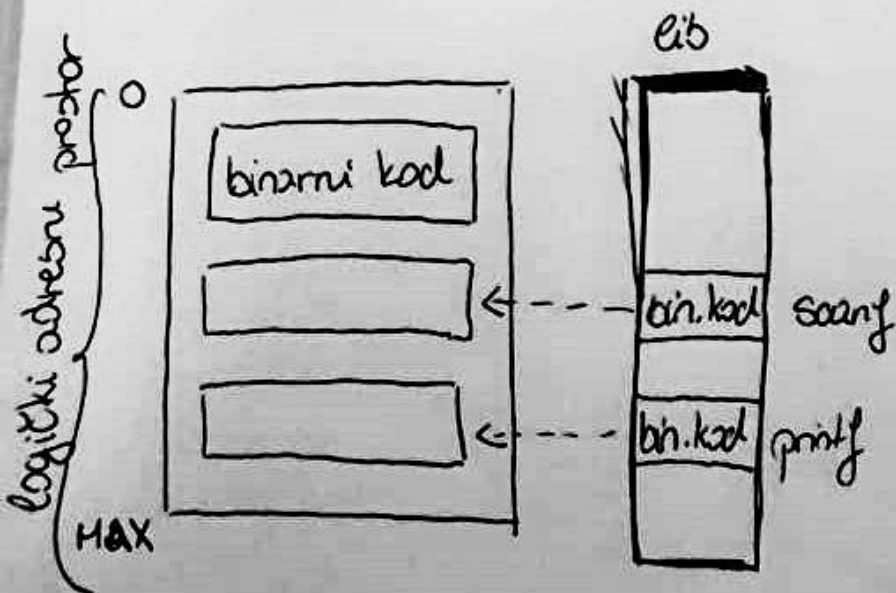
main	ok	
scanf	unknown	
printf	unknown	

binarni kod

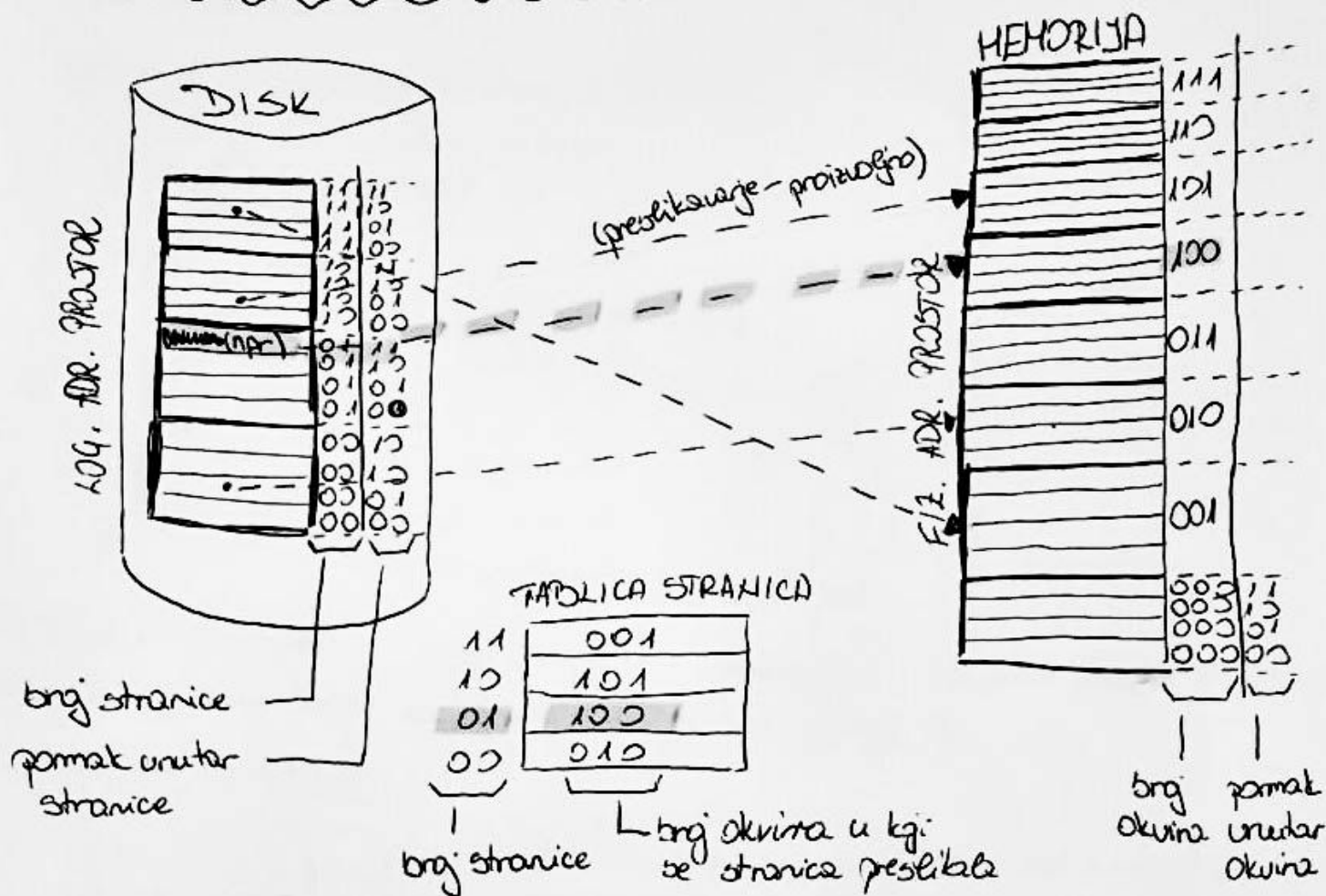
Nakon što su kompilirani svi moduli, sledi njihov povezivanje (LINK):

link program.o + lib / program.exe  
(neke dat.)

Linker koristi tablicu simbola da pronade nepoznate funkcije; u primeru to su `scanf` i `printf`.

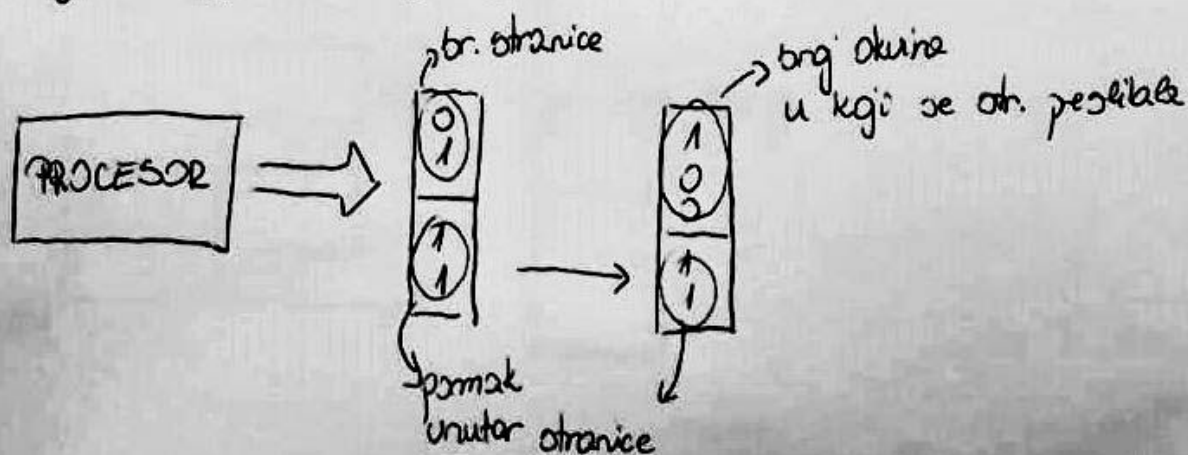


## 2. DODJELA MEMORIJE PO STRANICAMA

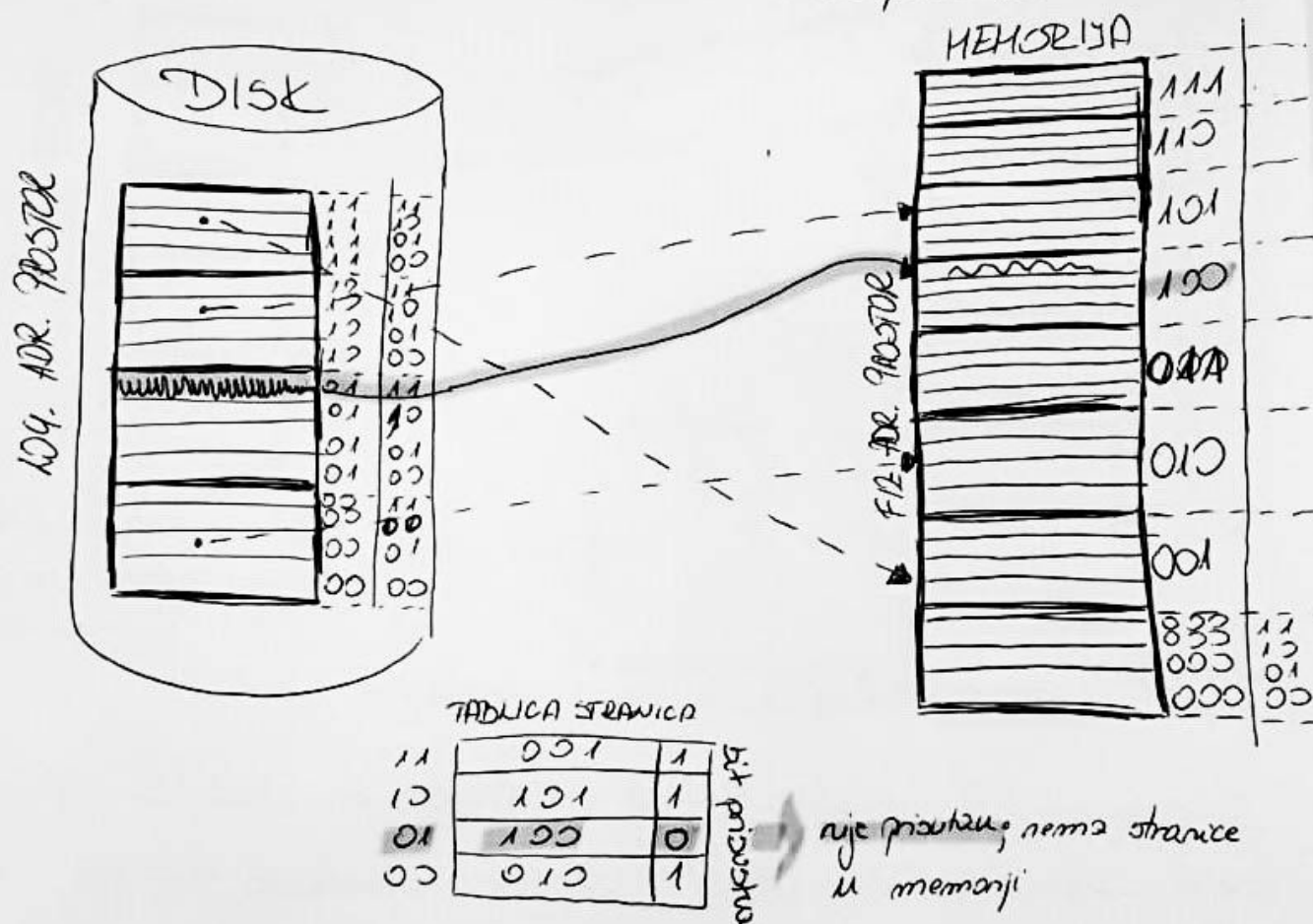


Programi i memorija se dijele na blokove veličine  $2^N$  - primjer  $N=2$ .  
 Blokovi log. adr. prostora - STRANICE; Blokovi fiz. adr. prostora - OKVIRI  
 Stranice se preslikavaju u određene slobodne okvire (proizvoljno)  
 Gdje je išla pojedina stranica, zapiše se u tablicu stranica.  
 Tablica stranica ima redaka koliko ima stranica.

Primjer: log. adr. koji generira procesor



3. VIRTUALNA MEMORIJA - proširene dočele memorije po stranicama na način da ne moraju sve stranice biti u memoriji. Prisustvo stranice u memoriji označava bit prisustva.



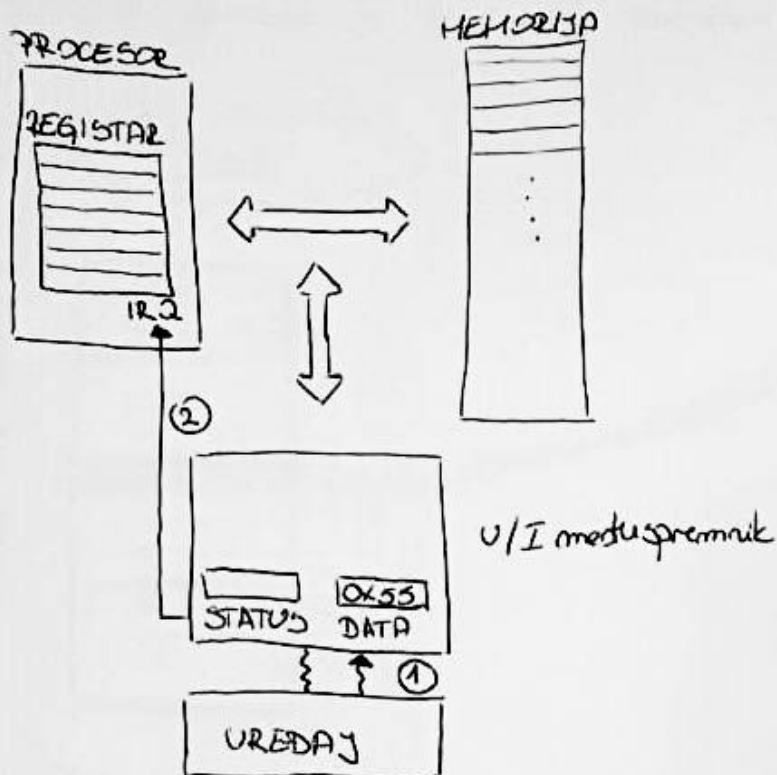
~~Bit prisustva~~ Bit prisustva nema, treba dovesti stranicu u memoriju.

Koraci:

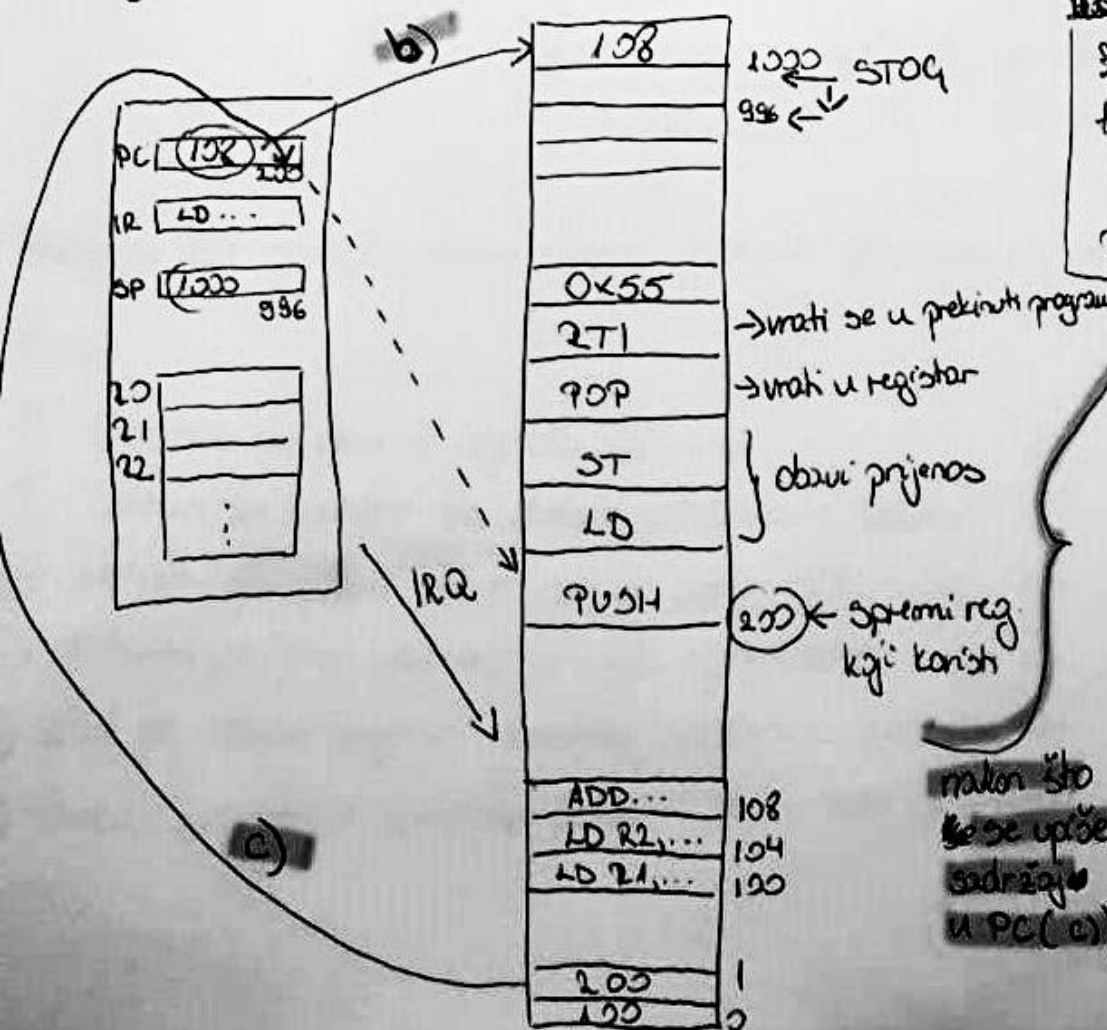
- ① Odabire se okvir u koji će stranica
- ② Postavi se zahtjev za dohvat stranice s diska
- ③ Aktivni ~~program~~ <sup>program</sup> stavi se u čeku disk
- ④ Aktivna se prvi proces iz reda pripravan
- ⑤ Kad se obavi prijem stranice, ažurira se tablica.
- ⑥ Proces se prebaci iz reda čeka disk u red pripravan.

#### 4. OBAVLAJANJE U/I OPERACIJA

① OBAVLAJANJE U/I OPERACIJA PO MOGUĆEM ZAHTEJU ZA PREKIDOM



#### Primjer



~~Obavljanje U/I operacija~~  
Zahtjev za prekid obično  
tijekom izvođenja naredbe  
LD R2,...

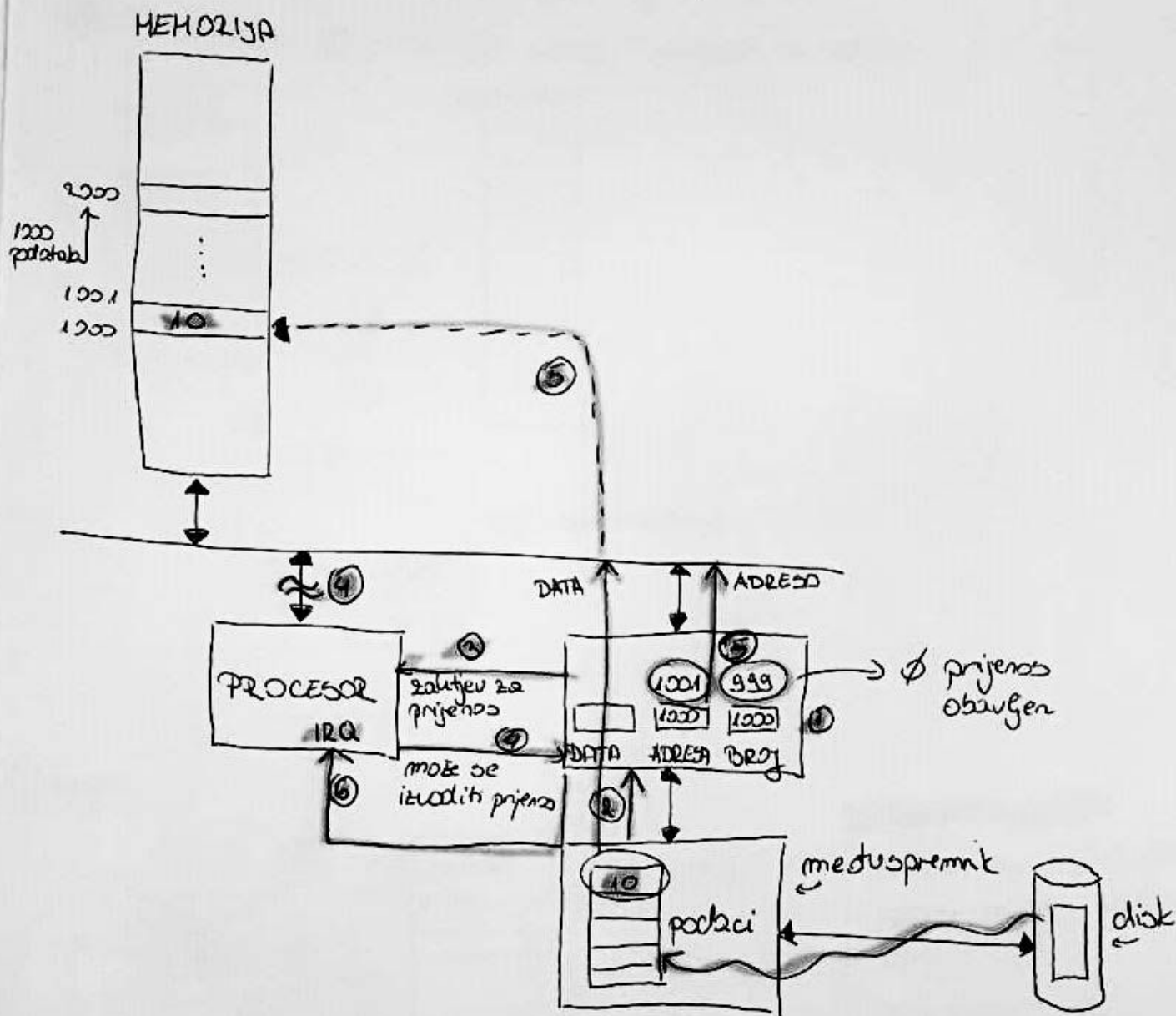
PROCESOR:

- odradi trenutnu naredbu do kraja **LD R2**,
- stavi PC na stog
- s adrese 4? (u adresi) upiše se sadržaj u PC

HARDWARE



## 4.5) DIREKTAN PRISTUP MEMORIJI (DMA)



① Inicijalizirati DMA sklop

ADRESA = 1000; BROJ = 1000

② Kontroler javi DMA - došli podaci

③ DMA postavi procesoru zahtjev za ~~prijenos~~ prijenos

④ Procesor, kad mu je zgodno, odgovori se od sabirnice i potvrdi DMA sklopu da može izvoditi prijenos

⑤ DMA sklop izvodi prijenos: adresa na adresnu sabirnicu, podatak na pod. sabirnicu, poveća se adresa ⇒ umanji se broj, ponavlja se dok broj = 0

⑥ DMA sklop, preko interupta, javi procesoru da je prijenos završio