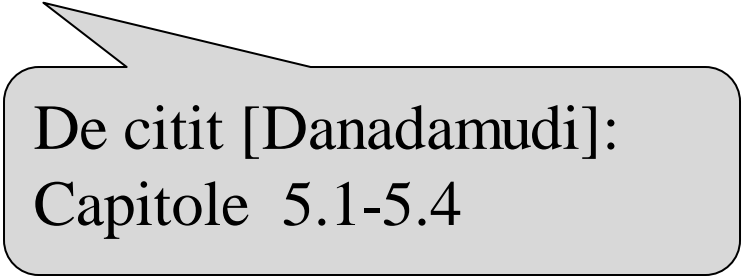

Stiva



De citit [Danadamudi]:
Capitole 5.1-5.4

Modificat: 22-Oct-23

Cuprins

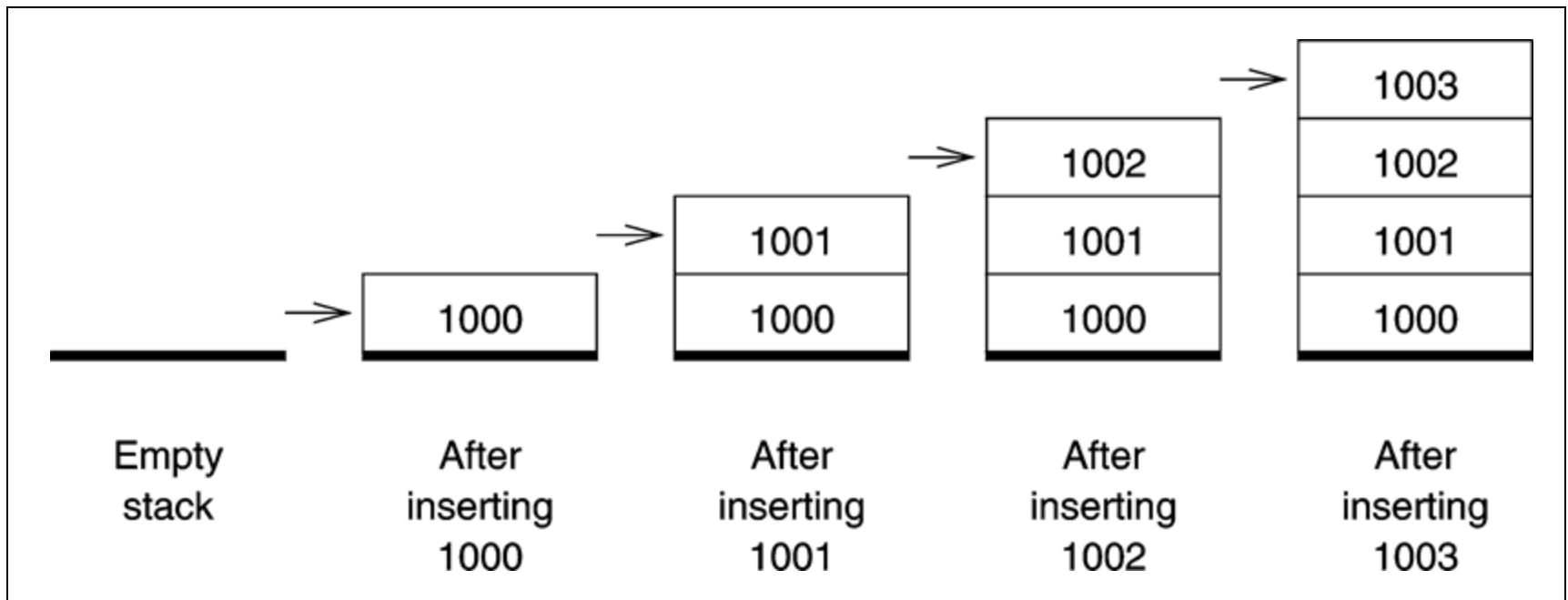
- Ce este stiva?
- Implementarea stivei pentru x86(Pentium)
- Instrucțiuni de lucru
- Utilizările stivei

Ce este stiva?

- Stiva este o coadă last-in-first-out (LIFO)
- Dacă vizualizăm stiva ca un vector de elemente, atunci inserția și ștergerea sunt restricționate la unul din capetele vectorului
- Numai elementul din vârful stivei (en. top-of-stack a.k.a TOS) este direct accesibil
- Structura implementează două operații de baza:
 - * push (inserție)
 - * pop (ștergere)

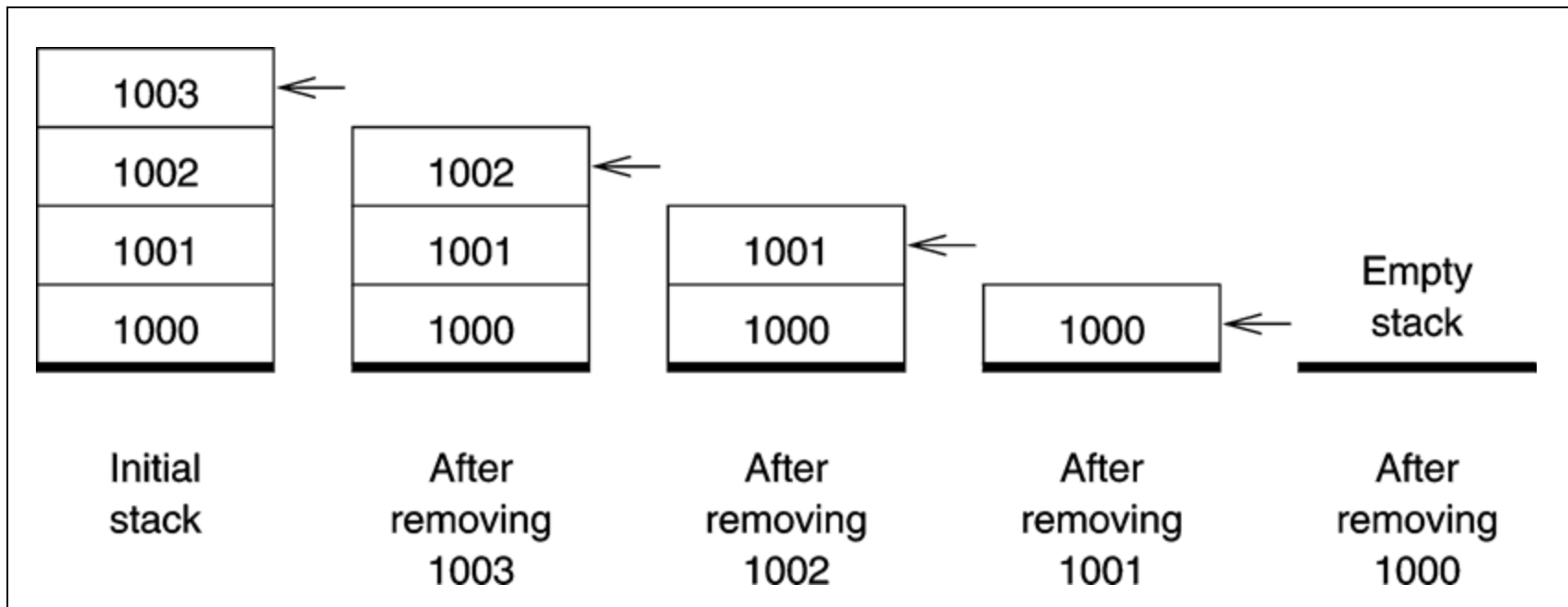
Ce este stiva? (cont'd)

- Exemplu
 - * Inserarea elementelor in stivă
 - » Săgeata pointează către vârful stivei



Ce este stiva? (cont'd)

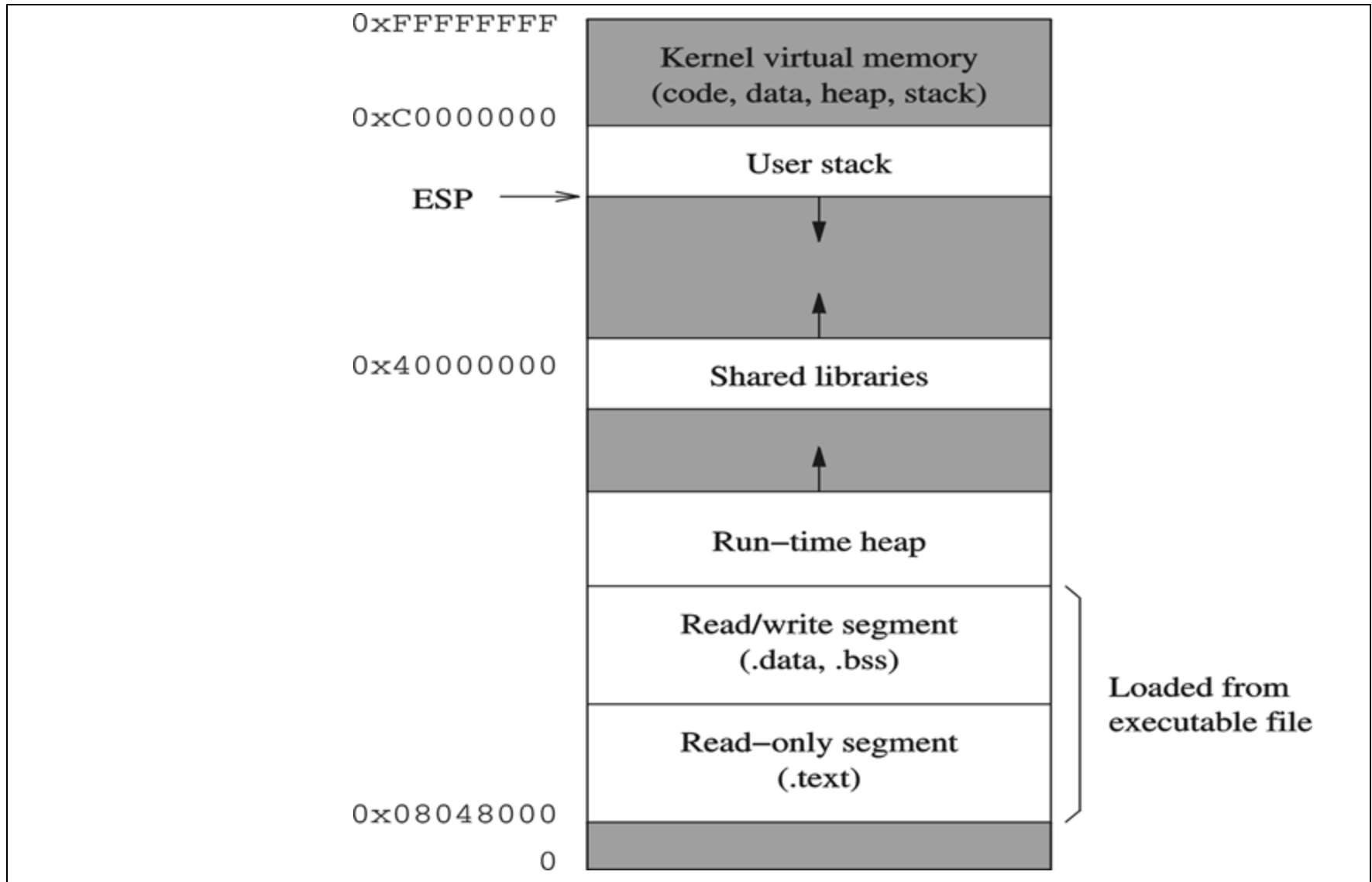
- Exemplu
 - * Ștergerea elementelor din stivă
 - » Săgeata pointează către vârful stivei



Implementarea stivei pentru x86

- La x86 este utilizat un registru segment rezervat:
 - * Registrul SS (Stack Segment) indică adresa de start a segmentului, iar registrul ESP (Extended Stack Segment) indica deplasamentul față de adresa start a vârfului stivei
 - * Impreuna SS:ESP indică vârful stivei
- Caracteristicile implementării pentru x86:
 1. Date de tip word/16-bit sau doubleword/32-bit
 2. Stiva crește **spre adrese mai mici “in jos”**
 3. TOS pointează către ultimul element introdus in stivă

Harta memoriei pentru un proces in Linux



Instrucțiuni de lucru cu stiva

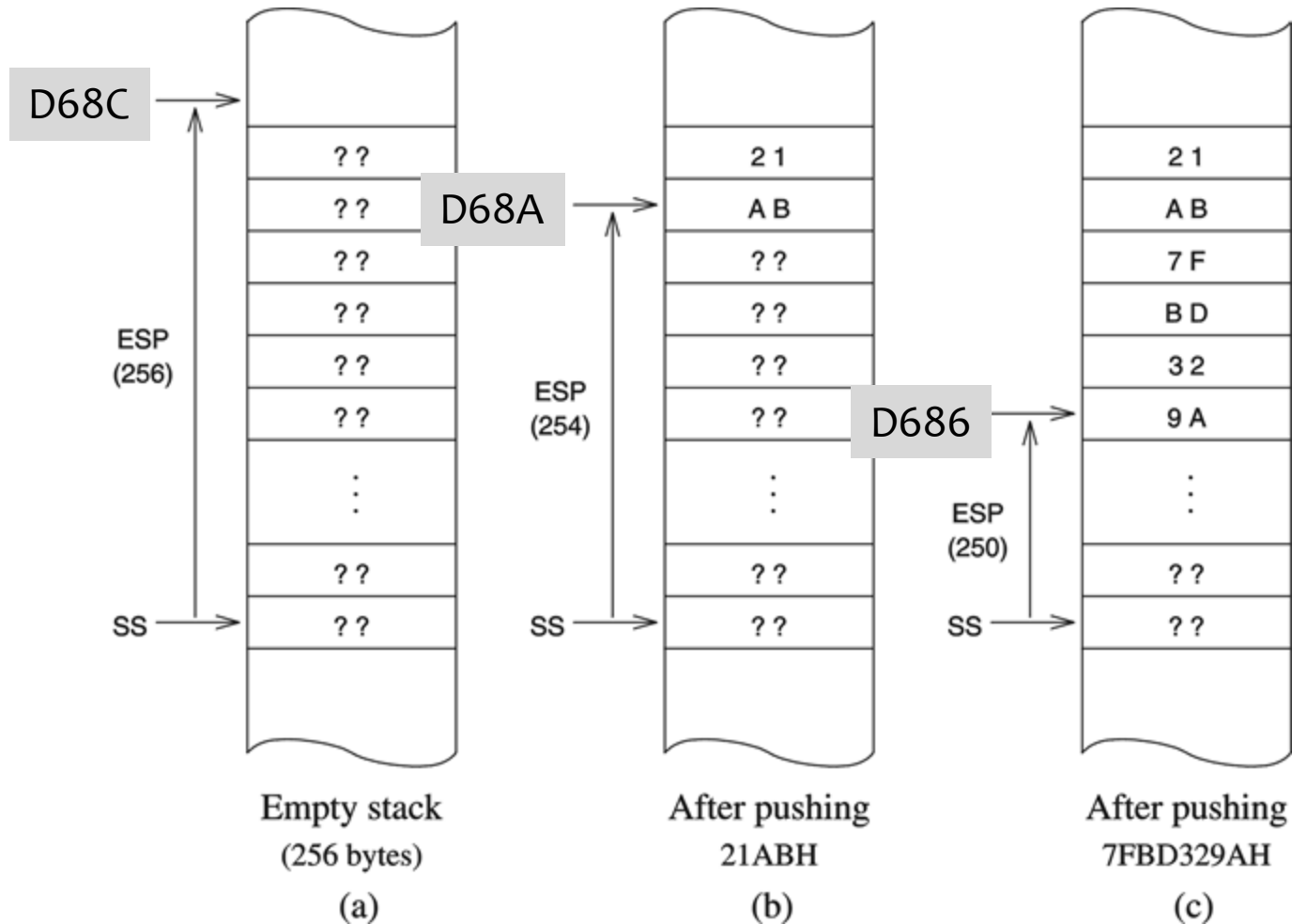
- x86 permite două instrucțiuni de **bază**:

push **sursa**

pop **destinație**

- **Sursa** și **destinația** pot fi
 - * registre de uz general 16- sau 32-bit (ex: ax sau eax)
 - * registru de segment (ex: DS, SS, CS, etc...)
 - * un word sau double word din memorie (ex: word [var+2])
- **source** în plus poate fi și o data imediată pentru instrucțiunea **push** de lungime 8, 16, sau 32 bit
 - *push 0xAB

Exemplu de lucru cu stiva pentru x86 - 1



Instrucțiuni de lucru cu stiva: Exemple

- Pentru o stivă goală următoarea secvență de instrucțiuni **push**

push word 21ABH

push 7FBD329AH

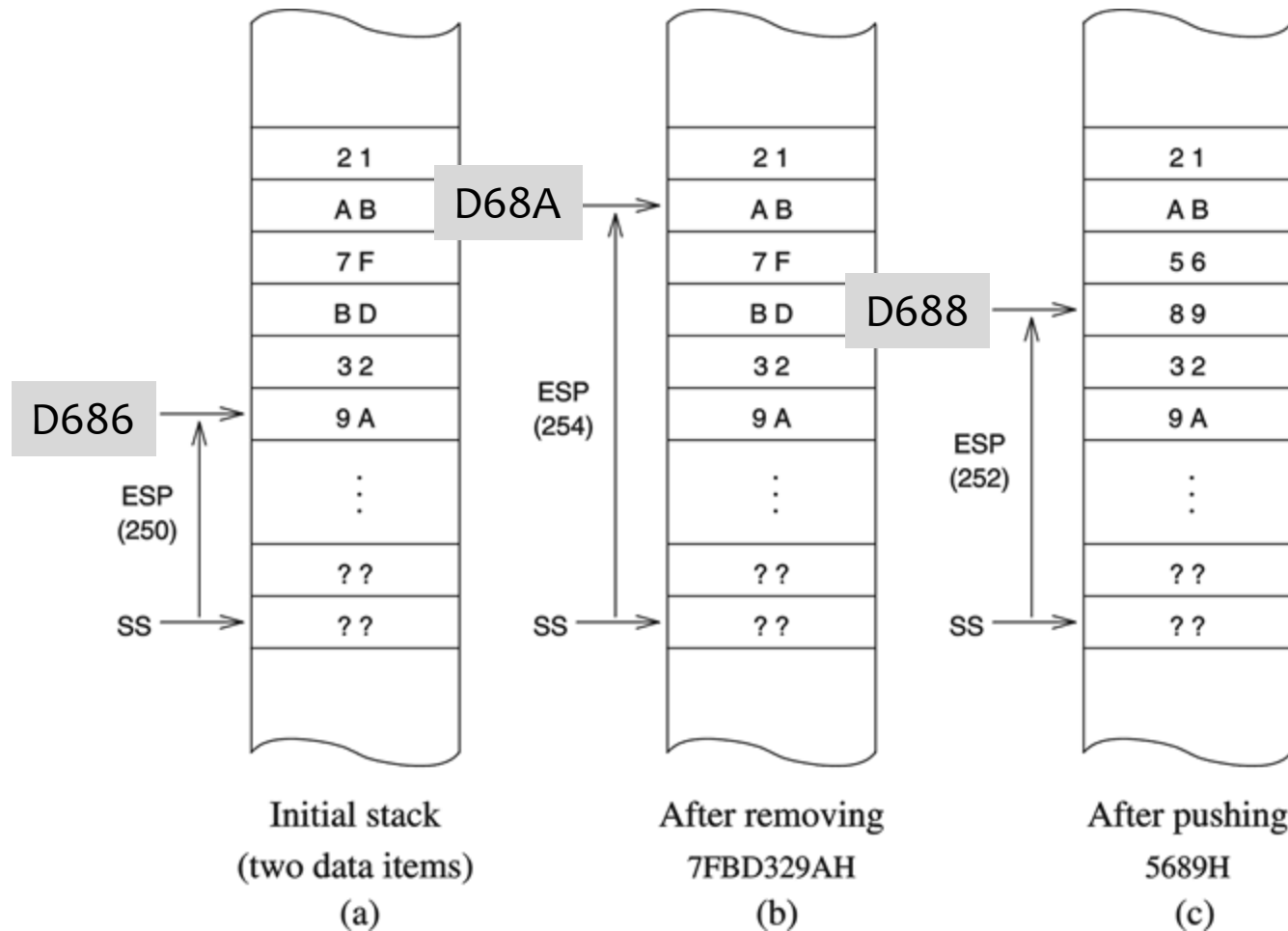
rezultă în starea prezentată la (c) în figura anterioară

- După execuția instrucțiunii:

pop EBX

Rezultă în starea stivei de la (b) din figura următoare, iar EBX va conține valoarea 7FBD329AH

Exemplu de lucru cu stiva pentru x86



Demo

```
z dd 0x1020304

; starea stivei
0xfffffd2ec f7de4b41 EBP+4
0xfffffd2e8 ffffd2ec EBP+0 <-- ESP
0xfffffd2e4 f7fa4000 EBP-4

push dword [z]
; în gdb x/4ub $esp răspunde 0xfffffd2e4: 4 3 2 1
; esp pointează la octetul LSB al ultimului element
; inserat

mov ebx, [esp]
pop eax
; ebx și eax conțin 0x1020304

mov esi, 0x0a0b0c0d
mov [esp-4], esi ; esp-4 pointează la primul spațiu liber
sub esp, 4 ; echivalent cu push esi
pop edi
; edi = esi
; stiva are aceeași stare ca la început
; sub esp e spațiu liber

0xfffffd2ec f7de4b41 EBP+4
0xfffffd2e8 ffffd2ec EBP+0 <-- ESP
0xfffffd2e4 0a0b0c0d EBP-4
```

Instrucțiuni adiționale

Operații de stiva asupra fanioanelor

- Instrucțiunile **push** si **pop** nu pot fi utilizate cu registrul de stare (EFLAGS)
- Doua instructiuni speciale in acest sens sunt:
 pushfd (push 32-bit flags)
 popfd (pop 32-bit flags)
- Operanzii nu sunt necesari
- Se folosesc **pushfw** and **popfw** pentru 16-bit (FLAGS)

Instructiuni aditionale

Operatii de stiva asupra la toti registrii de uz general

- Instrucțiunile **pushad** si **popad** sunt folosite pentru a salva si a restaura 8 registre de uz general
EAX, ECX, EDX, EBX, ESP, EBP, ESI, and EDI
- **pushad** executa o operație de push pentru fiecare din registrii anteriori în ordinea data (EAX primul și EDI ultimul)
- **popad** restaureaza toti registrii exceptand registrul ESP
- Se folosesc **pushaw** si **popaw** pentru a executa aceeași operație pentru registre la nivel de 16-bit (AX primul și DI ultimul)

Utilizările stivei

- Trei utilizari de baza:
 1. Stocarea datelor temporare
 2. Transferarea controlului
 3. Transmiterea parametrilor

1. Stocarea datelor temporare

Exemplu: Inter-schimbarea variabilelor **value1** si **value2** poate fi realizata utilizând stiva pentru a salva datele temporare

push	value1
push	value2
pop	value1
pop	value2

Utilizarea stivei (cont'd)

- Des utilizata pentru eliberarea unor registre

```
;salveaza EAX & EBX pe stiva  
push    EAX  
push    EBX  
;EAX si EBX pot fi acum folositi  
mov     EAX,value1  
mov     EBX,value2  
mov     value1,EBX  
mov     value2,EAX  
;restaureaza EAX & EBX din stiva  
pop     EBX  
pop     EAX  
. . .
```


Utilizarile stivei (cont'd)

2. Transferarea Controlului

- Pentru proceduri și întreruperi adresa de retur este salvata pe stiva
- Discuția pentru apelul procedurilor va clarifica în detaliu acesta utilizare

3. Transmiterea Parametrilor

- Stiva este extensiv utilizata pentru transmiterea parametrilor către proceduri
- Discuția de mai târziu va arata cum se realizează acest proces