המרכז
לחינוך סייבר
CYBER EDUCATION CENTER

Assembly for Reverse Engineering

String Operations

Barak Gonen

ESI, EDI, MOVSB

המרכז לחינוך סייבר
CYBER EDUCATION CENTER

Extended Source Index

Extended Destination Index

MOVS: MOVSB – Byte, MOVSW – Word, MOVSD - DWord

Try it:

```
mov     esi, msg1
mov     edi, msg2
movsb
```

REP

המרכז
לחינוך סייבר
CYBER EDUCATION CENTER

```
mov     ecx, [msg_len]
mov     esi, msg1
mov     edi, msg2
rep     movsb
```

CLD / STD

Direction Flag (DF) sets the direction of progress-

- DF = 0: ESI = ESI+1, EDI=EDI+1
- DF = 1: ESI = ESI-1, EDI=EDI-1

CLD – Clear flag

STD – Set flag

LODS

המרכז לחינוך סייבר
CYBER EDUCATION CENTER

Same as MOVS, but to a register

Run and use OllyDbg:

```
xor     eax, eax
mov     esi, msg2
lodsb
call    print_eax
lodsw
call    print_eax
lodsd
call    print_eax
```

STOS

המרכז לחינוך סייבר
CYBER EDUCATION CENTER

Opposite of LODS

Note: EDI progresses

Exercise:

- Use STOS to store ‘Assembly’ into an empty string
- Use as few instructions as possible!

Other Forms of REP

המרכז לחינוך סייבר
CYBER EDUCATION CENTER

Form	Stop Condition 1	Stop Condition 2
REP	ECX = 0	NONE
REPE / REPZ	ECX = 0	ZF = 0
REPNE / REPNZ	ECX = 0	ZF = 1

CMPS, SCAS

המרכז
לחינוך סייבר
CYBER EDUCATION CENTER

Self study ☺

Exercise: strcmp.asm