

**Registrene** er spesielle heltallsvariabler som er ekstra tett koblet til regneenheten. Vi skal bruke disse registrene:

%EAX %ECX %EDX %EBP %ESP

<u>func:</u>	<u>movl</u>	<u>\$0,%eax</u>	<u># Initier til 0.</u>
Navnelapp	Instruksjon	Parametre	Kommentar

**Parametre** angir data til instruksjonen; antallet avhenger av instruksjonen.

Vi vil bruke disse parametrene:

**%EAX** er et register.

**\$17** er en tallkonstant.

**f** er navnet på en prosedyre eller en funksjon.

**8(%ESP)** angir en variabel eller en parameter.

**Kommentarer** ignoreres.

.globl xxx	Navnet xxx skal være kjent utenfor filen
movl <v <sub>1</sub> >, <v <sub>2</sub> >	Flytt <v <sub>1</sub> > til <v <sub>2</sub> >.
cdq	Omform 32-bits %EAX til 64-bits %EDX:%EAX.
leal <v <sub>1</sub> >, <v <sub>2</sub> >	Flytt <v <sub>1</sub> >s <i>adresse</i> til <v <sub>2</sub> >.
pushl <v>	Legg <v> på stakken.
popl <v>	Fjern toppen av stakken og legg verdien i <v>.
negl <v>	Skift fortegn på <v>.
addl <v <sub>1</sub> >, <v <sub>2</sub> >	Adder <v <sub>1</sub> > til <v <sub>2</sub> >.
subl <v <sub>1</sub> >, <v <sub>2</sub> >	Subtraher <v <sub>1</sub> > fra <v <sub>2</sub> >.
imull <v <sub>1</sub> >, <v <sub>2</sub> >	Multipliser <v <sub>1</sub> > med <v <sub>2</sub> >.
idivl <v>	Del %EDX:%EAX med <v>; svar i %EAX; rest i %EDX.
andl <v <sub>1</sub> >, <v <sub>2</sub> >	Logisk AND.
orl <v <sub>1</sub> >, <v <sub>2</sub> >	Logisk OR.
xorl <v <sub>1</sub> >, <v <sub>2</sub> >	Logisk XOR.
call <lab>	Kall funksjon/prosedyre i <lab>.
enter \$<n <sub>1</sub> >, \$<n <sub>2</sub> >	Start funksjon/prosedyre på blokknivå <n <sub>2</sub> > med <n <sub>1</sub> > byte lokale variabler.
leave	Rydd opp når funksjonen/prosedyren er ferdig.
ret	Returner fra funksjon/prosedyre.
cmpl <v <sub>1</sub> >, <v <sub>2</sub> >	Sammenligning <v <sub>1</sub> > og <v <sub>2</sub> >.
jmp <lab>	Hopp til <lab>.
je <lab>	Hopp til <lab> hvis =.
sete <v>	Sett <v>=1 om =, ellers <v>=0.
setne <v>	Sett <v>=1 om ≠, ellers <v>=0.
setl <v>	Sett <v>=1 om <, ellers <v>=0.
setle <v>	Sett <v>=1 om ≤, ellers <v>=0.
setg <v>	Sett <v>=1 om >, ellers <v>=0.
setge <v>	Sett <v>=1 om ≥, ellers <v>=0.

**Tabell: x86-instruksjoner** brukt i prosjektet. Følgende symboler er brukt i tabellen:

<v> kan være en konstant («\$17»),

et register («%EAX»),

en lokal variabel («-4(%EBP)») eller

en parameter («8(%EBP)').

<n> er en heltallskonstant.

<lab> er en merkelapp som angir en minnelokasjon.