

| | |
|---------------------------|--|
| %include "gt.asm" | Het begin van een oefening |
| covar inlijding | |
| eax | Is een register van 32bit om bewerkingen in uitvoeren |
| mov | Kopiëren |
| add | Optellen bij een register |
| uit | Om een getal op het scherm te laten printen |
| imul dword | Om eax te vermenigvuldigen met een getal |
| idiv | Om eax te delen met een getal en de rest word in EDX gezet |
| edx, ebx, ecx... | registers om bewerkingen in uitvoeren (ecx: count register) |
| jmp (naam) | Om een sprong uitvoeren (jump bevel) |
| Inv | Om een getal aan de gebruiker te vragen |
| ZF, SF, OF | ZF: null-vlag SF: teken-vlag OF: overloop-vlag |
| cmp | Om 2 getallen te vergelijken (er moet 1 een register zijn) |
| je, jle, jl, jg, jge, jne | j=jump e=equal l=less g=greater n=not |
| eip | De bevelenteller |
| loop | Automatisch jump bevel (aantalkeer moet in ecx gezet worden) |
| DD | Define dubbelwoord |
| DB | Define byte |
| movsb | Een karakter kopiëren |
| rep movsb | Een string kopiëren |
| cld std | behandel richting instellen (richtingsvlag) |
| lodsb | Een karakter kopiëren naar het geheugen van de cpu |
| rep lodsb | Een karakter meermaals kopiëren naar het geheugen van de cpu |
| esi | De verplaatsing ven de te kopiëren byte |
| edi | Het kopieer doel van de inhoud van esi |
| stosb | het kopieën van een byte in (AL) naar de source in(EDI) |
| lodsb | het kopieën van een string van source (ESI) naar (AL) |
| outarea | Om een zint in op te slaan en bestaat uit (resb 70 + DB 0Dh; 0Ah) |
| schrijf | Om wat in outarea staat af te drukken en een nieuwe lijn te starten |
| openuit | Hierdoor word een uitvoer bestand gevormd |
| inarea | Om een zint in opteslaan die uit een invoer bestand gelezen word |
| lees | Kopieert een regel uit het invoer bestand |
| tekstbin | Om tekst naar een getal om te zetten (verplaatsing in ESI) (aantal bytes in ECX) (resultaat in EAX) |

Deze commando samenvatting is tot p107