

Assembler x86:

krótki kurs

Dominik M. Matusiak

Assembler x86: krótki kurs

dr Dominik M. Matusiak

d.matusiak@op.pl

Wydawnictwo Eclipse, 2026 r., Zgierz

ISBN 978-83-975336-0-8

ISBN 978-83-975336-0-8



9 788397 533608

Wstęp

Assembler to język programowania niskiego poziomu (język w którym jednej operacji odpowiada najczęściej jedna operacja procesora). Języki niskiego poziomu są ściśle powiązane z daną architekturą sprzętową. Polecenia w assemblerze nazywane są mnemonikami.

Niniejszy podręcznik jest jedynie zwięzłym wprowadzeniem do assemblera (Intel x86). Bardziej zaawansowane kwestie omawiane są w innych książkach.

Zastosowania assemblera

Obecnie assembler ma mniejsze znaczenie niż kiedyś (względnie popularny był w erze systemu DOS) ze względu na rozwój języków wysokiego poziomu i złożonych systemów operacyjnych (np. Windows, Linux).

Assembler wykorzystywany jest m.in. przy tworzeniu systemów operacyjnych (szczególnie jądra, sterowników), bootloaderów, wirusów, emulatorów, usuwaniu błędów z oprogramowania (debugowanie), crackowaniu, hackowaniu (np. tzw. ROM hacki) i optymalizacji oprogramowania, analizy działania oprogramowania (reverse engineering, szczególnie malware), jako wstawki assemblerowe w kompilowanych językach wysokiego poziomu (np. w C), przy programowaniu systemów wbudowanych.

Wstawki assemblerowe

Są to fragmenty kodu w assemblerze wstawiane do kodu napisanego w języku wysokiego poziomu (kompilowanym). Przykład programu napisanego w C (ze wstawkami asm), który sumuje dwie liczby:

```
#include <stdio.h>
```

```
int main() {  
    int a = 10, b = 20, suma;  
    __asm__ (  
        "addl %%ebx, %%eax;" // eax = eax + ebx  
        : "=a" (suma)      // suma = eax  
        : "a" (a), "b" (b) // a w eax, b w ebx  
    );  
    printf("Suma: %d\n", suma);  
}
```