

Podstawowe komendy terminala:

Cls – [wyczyść terminal](#)

Cd ścieżka – [chodzenie po plikach](#)

Nasm -o plik_docelowy.com plik.asm – [kompilacja pliku.asm](#)

Plik_docelowy.com – [odpalanie pliku](#)

Podstawowe komendy Assembler:

Org 100h – [to musi być na początku](#)

mov ax, 4C00h – [to mówi konsoli, że kończymy \(bez tego może się wieszać\)](#)

int 21h – [zatwierdź koniec](#)

coś: - [etykieta, pomagająca odnaleźć się w kodzie \(uwaga na spacje i polskie literki\)](#)

start: - [etykieta rozpoczęcia się właściwego kodu](#)

jmp etykieta – [przeskocz bezpośrednio do pewnej etykiety](#)

mov rejestr, komenda – [zapytanie wrzucić do rejestru](#)

int 21h – [wykonaj](#)

Rejestry:

Rejestry_Danych:

AX (Accumulator Register) – [używany przy obliczeniach](#)

BX (Base R.) – [używany przy adresowaniu](#)

CX (Count R.) – [licznik](#)

DX (Data R.) – [operacje wejścia/wyjścia](#)

Rejestry_Wskaźnikowe:

IP (Instruction Pointer) – [zawiera adresy aktualnie wykonywanych operacji](#)

SP (Stack P.) – [wskaźnik stosu](#)

BP (Base P.) – [do adresowania pamięci](#)

Rejestry_Indeksowe:

SI (Source Index) – [służy do indeksowania pamięci](#)

DI (Destination I.) – [odwrotność SI](#)

Rejestry_Segmentowe:

CS (Code Segment) – [zawiera instrukcję do wykonania, posiada adres początkowy kodu](#)

DS (Data S.) – [zawiera dane, przechowuje adres segmentu danych](#)

SS (Stack S.) – [segment stosu](#)

Mov ax, bx – [adresowanie rejestrowe](#)

Mov ax, 1 – [adresowanie bezpośrednie](#)

Mov ax, [102h] – [adresowanie pośrednie](#)

Mov ax, [di] – [adresowanie rejestrowe pośrednie](#)

Przerwania:

Int 21h – [przerwanie DOS. Najważniejsze.](#)

Int 10h – [przerwanie karty VGA. Wykorzystywany przy zmianie ustawień i lokalizacji wyświetlania.](#)

Int 16h – [przerwanie klawiatury.](#)

Zmienne:

Typ zmiennych:

DB – [zmienne bitowe \(byte, char\). Tego używamy](#)

DW – [zmienne słowne \(int\)](#)

DD – [zmienne dwusłowne \(long int, 2 bajty\)](#)

DF – [zm. Trzysłowne \(real\)](#)

DQ – [zm. Czterosłowne \(double\)](#)

DT – [zm. Pięciosłowne \(extended\)](#)

Nazwa_Zmiennej TYP wartość – [deklarowanie nowej zmiennej](#)

TEKST db „Cześć! \$”

CIAG_LICZB db 1, 2, 3, 4

ZNAKI db „abc”, 20, „xyz”

Mov DX, TEKST – [przykładowe używanie zmiennej](#)

\$ - [mówi assemblerowi w którym miejscu kończy się napis](#)

Arytmetyka:

ADD x, y – $x = x + y$

SUB x, y – $x = x - y$

MUL BX – mnożenie ($BX * AX = DX AX$)

DIV BL – dzielenie ($AX/BL = AL AH$)

INC x – $x = x + 1$

DEC x – $x = x - 1$

Porównania:

Cmp Ax, Bx – Porównaj rejestry Ax do Bx

Jl etykieta – jeżeli $Ax < Bx$ to skocz do etykiety

Jle etykieta - jeżeli $Ax \leq Bx$ to skocz do etykiety

Jg etykieta - jeżeli $Ax > Bx$ to skocz do etykiety

Jge etykieta - jeżeli $Ax \geq Bx$ to skocz do etykiety

Je etykieta - jeżeli $Ax == Bx$ to skocz do etykiety

Jne etykieta - jeżeli $Ax \neq Bx$ to skocz do etykiety

Funkcje:

Call etykieta – wywołanie funkcji

Ret – to musi się znaleźć na końcu etykiety, aby była ona funkcją

Dodatkowo:

Alt + Enter – full screen

<http://www.ctyme.com/intr/int.htm> - dokumentacja

<http://aragorn.pb.bialystok.pl/~teodora/AK/W2.pdf> - nauka