# Architektura Systemów Komputerowych

Laboratorium 8

Asembler dla procesora Intel 80x86

Opracował:
mgr inż. Leszek Ciopiński

---

**Wstęp:**

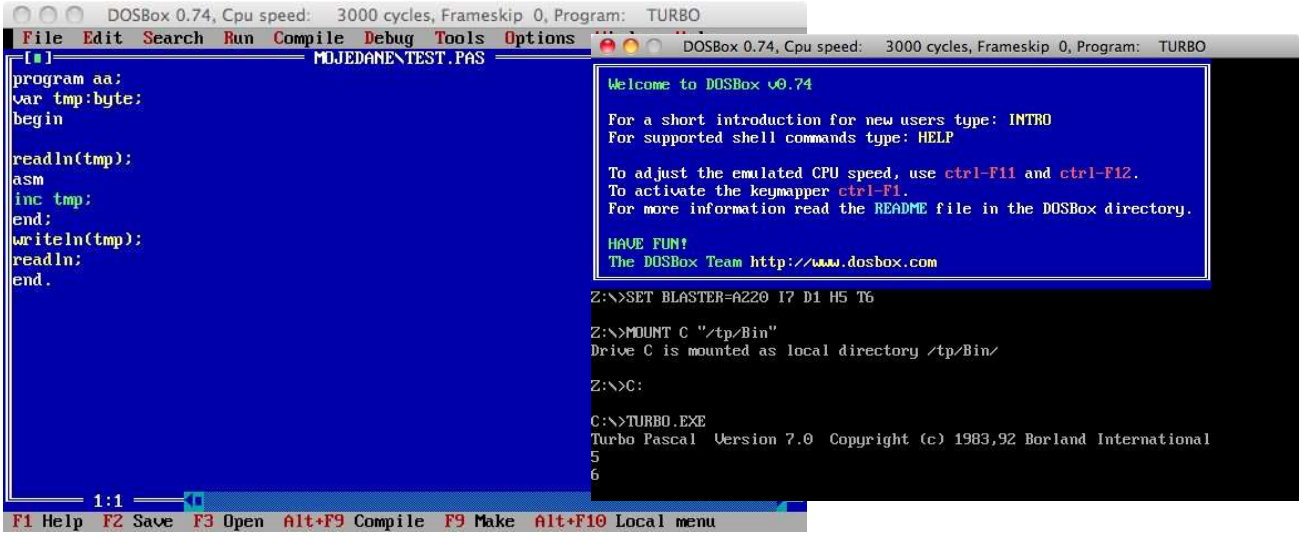
Informacje ogólne:

Z dołączonego wyciągu z pl.wikibooks.org proszę zapoznać się z następującymi rozdziałami:

- Asembler X86/Wstęp, strona 3
- Asembler X86/Architektura, strony 16 – 21
- Pamięć, strony 25 - 26

Wstawka asemblerowa w Turbo Pascalu:

W środowisku Borland Turbo Pascal możemy umieszczać krótkie programy napisane w języku asembler bezpośrednio w kodzie programu przy pomocy polecenia *asm*. W każdym miejscu wstawki mamy dostęp do zmiennych Pascala. Wstawkę kończymy poleceniem *end*.




Zmienne dostępne w wstawce

Automatyczne wewnątrz wstawki możemy używać zmiennych skojarzonych z rejestrami. Podstawowymi z nich są: AX (AH, AL), BX (BH, BL), CX (CH, CL) i DX (DH, DL). Ponadto dostępne są rejestry służące do adresowania pamięci.

Etykiety

W języku asembler możliwe jest wykonanie skoku w dowolne miejsce programu. W celu

oznaczenia danego miejsca stosuje się tzw. etykiety, które na etapie kompilacji tłumaczone są na konkretne adresy. W celu wprowadzenia etykiety stosuje się zapis:

@nazwa:

Aby rozpocząć wykonywanie kodu bezpośrednio za etykietą, należy wykonać skok do niej przy pomocy polecenia *goto*:

goto @nazwa

Podstawowe operacje

W poniższych przykładach symbole z1 i z2 oznaczają, że możliwe jest podstawienie w dane miejsce dowolnego rejestru. Dodatkowo w miejsce rejestru z2 może być podstawiona stała wartość. Jeśli dana operacja wymaga konkretnego rejestru, wówczas zostanie podana jego nazwa jako parametr.

| | |
|---|---|
| mov z1, z2 | zawartość rejestru z2 kopiowana jest do z1 |
| add z1, z2 | wykonywana jest operacja z1:=z1+z2 |
| adc z1, z2 | wykonywana jest operacja z1:=z1+z2+C. Wartość C, to bit przeniesienia. Jeśli w poprzedniej operacji sumowania wystąpiło przeniesienie, to bit C ustawiony jest na 1. W przeciwnym przypadku wynosi 0. |
| loop @etykieta | skok do @etykiety, jeśli wartość rejestru CX > 0. Następnie dekrementuje wartość tego rejestru. Jeśli warunek nie jest spełniony, wykonywana jest kolejna instrukcja – brak skoku. Rozkaz używany do budowy pętli |
| mul z2 | wykonywana jest operacja AX:=AL*z2[7..0] |

Przerwanie programowe

W środowisku asemblerowym w celu skorzystania z funkcji dostarczonych przez system operacyjny, konieczne jest wywołanie przerwania programowego. Jednym z najczęściej stosowanych jest:

INT 21h

Wynik działania tego przerwania zależy od wartości zapisanej w rejestrze AH.

AH = 2Ah : w rejestrze CX dostajemy bieżący rok, w DH - miesiąc, a w DL - dzień miesiąca. Ponadto, w AL dostajemy numer dnia tygodnia (0 oznacza niedzielę)
AH = 2Ch : w rejestrze CH dostajemy bieżącą godzinę, w CL - minutę, a w DH - sekundę.

**Zadania:**

1. Wczytaj dowolną liczbę dodatnią. Następnie w wstawce asemblerowej wykonaj jej inkrementację i podniesienie do sześcianu. Po opuszczeniu wstawki wykonaj wyświetlenie obliczonej wartości. *(2 punkty)*
2. Wczytaj dwie liczby 16 bitowe, a następnie wykonaj ich dodawanie w asemblerze. Wyświetl wynik w taki sposób, aby zawsze był poprawny (nawet jeśli dojdzie do przekroczenia zakresu). Użyj wyłącznie typy liczb nieujemnych. *(2 punkty)*
3. Wewnątrz wstawki asemblerowej wykonaj obliczenie w pętli 10. elementu ciągu liczb Fibonacciego. Powtórz obliczenia poza wstawką i wyświetl uzyskane wyniki. *(3 punkty)*
4. Wewnątrz wstawki asemblerowej wykorzystaj przerwanie programowe w celu odczytania informacji o aktualnym czasie. Wynik należy wyświetlić w formacie: dd-mm-rrrr hh:mm:ss. *(3 punkty)*