

Ćwiczenie 2. Instrukcje warunkowe, wyboru i pętli

Celem ćwiczenia jest zapoznanie się ze sposobem realizacji podstawowych instrukcji warunkowych oraz pętli w języku asemblera procesora 8086.

Zagadnienia do przygotowania

- składnia i semantyka instrukcji `if...then...else` w języku C
- budowa instrukcji wyboru `switch`
- składnia i semantyka (schematy blokowe) pętli `while...` i `for...`
- instrukcje arytmetyczne i porównania procesora 8086
- interpretacja znaczników w słowie stanu procesora
- rozkazy skoków warunkowych i bezwarunkowych (względnych i bezwzględnych).

Przebieg ćwiczenia

- przygotować schemat blokowy programu podanego w załączniku w języku asemblera
- przeprowadzić jego kompilację i linkowanie za pomocą pakietu Borland C 3.1
- sprawdzić poprawność działania programu za pomocą Turbo Debuggera, zwrócić szczególną uwagę na prawidłowość warunku opuszczenia pętli i zawartość tablic.

Warunki zaliczenia ćwiczenia

Zaliczenie ćwiczenia polega na demonstracji działania programu prowadzącemu.

Sprawozdanie

Sprawozdanie powinno zawierać wydruk programu z obszernymi komentarzami oraz schematy blokowe wszystkich instrukcji strukturalnych.

Literatura

Kernigham, Ritchie, *Język C*, WNT 1989

Dokumentacja pakietu Borland C 3.1

Wróbel Eugeniusz, *Asembler 8086/88*

Wacławek Roland, *ABC Asemblera*

Scanlon Leo J., *Assembler 8086/8088/80286*

Kruk Stanisław, *Język Assembler dla początkujących*

Nawrocki J. R., *Programowanie komputerów IBM PC w języku Asemblera metodą systematyczną*

Syck Gary, *Turbo Assembler : biblia użytkownika*

Laboratorium: Programowanie niskopoziomowe

Deklaracje zmiennych

```
char napis1[4] = "ala";
char napis2[4]; char napis3[4]; char napis4[4];

char * ptr1,* ptr2;

int i,j;
```

Instrukcje programu

```
void main(void)
{
ptr1=napis1; ptr2=napis2; ptr3=napis3;

/* kopiowanie lancuchow poprzez zmienne indeksy
   wskazowka zastosuj instrukcje typu
       MOV napis2[BX],AL */
for(i=0,j=0; napis1[i]!=0; i++,j++)
{
    napis2[j]=napis1[i];
}
/* czy wartosci koncowe zmiennych indeksowych i,j sa
   zgodne ze standardem jezyka C ? */

/* kopiowanie lancuchow poprzez wskazniki */
while(*ptr3++=*ptr2++);
/* wskazówka wykorzystaj efektywnie instrukcje MOV [BX],AL */

/* instrukcja wyboru switch */

i=2;
/* wskazowka: w programie zastosuj instrukcje
...
    CMP AX,1
    JE case1
    CMP AX,2
    JE case2
...
case1:...
...
case2:

Zastanow sie, gdzie wstawic odpowiedniki break?
Jakie znaczenie w tej konstrukcji ma default?
Jaka wartosc ma zmienna j po wykonaniu instrukcji switch? */

i=0;
switch (i) {
    case 0: j=napis1[0];
    case 1: j=napis1[1]; break;
    case 2: j=napis1[2];break;
    default: j='a';
            }
}
```