

Numele si prenumele

Grupa

1. Sa se scrie urmatoarele functii, in limbaj de asamblare 8086:

a) `void compute(int a[], int b[], int c[], int n, int id)`

care prelucraza tablourile **a** si **b**. Operatiile posibile sint: adunare , scadere si impartire. Rezultatul se scrie in tabloul **c**. Toate tablourile au **n** elemente. Parametrul **id** specifica functia de prelucrare (0 – adunare, 1 – inmultire si 2 impartire). Se vor implementa structuri de control *for* si *switch*.

Adresele initiale ale tablourilor se vor stoca astfel: pentru tabloul **a** in SI, pentru tabloul **b** in DI si pentru tabloul **c** in BX. Numarul de elemente **n** este stocat in registrul CL si identificatorul functiei **id** este plasat in CH.

b) `void bit_prel (int w[], int b, int p)`

care prelucraza pe biti un tablou de cuvinte **w** (terminat cu 0x0000): seteaza bitul **b** din fiecare cuvant **w[i]**.

Adresa initiala a tabloului **w** este stocata in BX si bitul **b** este stocat in CL. Parametrul **p** (stacat in CH) indica daca prelucrarea este setare (**p=1**) sau resetare (**p=0**). Se vor implementa structuri de control if si while

c) `int bit_id (int w[], int b, int i)`

care prelucraza pe biti un tablou de cuvinte **w** (terminat cu 0x0000): intoarce in AL valoarea bitului **b** din cuvantul **w[i]**. Adresa initiala a tabloului **w** este stocata in BX, bitul **b** este stocat in CL, parametrul **i** este stocat in CH. Se va implementa o structura de control *do until* .

Observatie: se pot utiliza secventele scrise la fisele 1 si 2.

Descrierea programului (pseudocod, explicatii)

Codul in limbaj de asamblare si rezultatele obtinute