

Seminar 7

Problema 1 – 3p – 26.01.2023

Se dă un șir ASCIIZ de secvențe de litere separate prin caracterul spațiu în modulul **a.asm**. Scrieți un program multimodul (asm+asm) care determină și afișează pe ecran secvențele de litere care au număr par de vocale și în același timp număr impar de consoane, și tipărește pe ecran în baza 16 numărul acestor secvențe, fără a utiliza specificatorii %x, %X. În acest scop, programul va apela o funcție denumită **procesare** definită în modulul **b.asm**, care primește ca parametru adresa de început a secvenței curente de prelucrat și verifică dacă secvența îndeplinește condițiile pentru afișare. Nu se va folosi nici o funcție de prelucrare de șiruri a limbajului C. **Explicați și detaliați abordarea algoritmului și mecanismele implicate. Justificați și comentați corespunzător textul sursă. Prezentați și explicați mecanismele de comunicare utilizate între cele două module dezvoltate.**

Exemplu: sir db 'Ana are mere Ada are fructul pasiunii Gigel are ananas Tudor are portocale si facem salata de fructe', 0

Numărul de secvențe de litere cu număr par de vocale și număr impar de consoane este 11 = 0Bh deci pe ecran se va afișa:

Ana are Ada are fructul Gigel are Tudor are portocale facem

B

Main.asm

0.25p - algoritm descriere în cuvinte

0.25p - schelet program + segment de date + comentarii

0.25p - parcurgere șir mare de litere (se punctează inclusiv faptul că după o secvență de litere continuă corect cu următoarea)

0.1p – are condiție de oprire și tratează corect și ultima secvență care se termină cu 0

0.15p - apel corect funcție procesare (cu parametrul corect)

0.25p - verificare dacă secvența curentă respectă condițiile în funcție de rezultatul funcției (e la alegerea lor ce și cum returnează, dar trebuie să verifice undeva nr par de vocale și nr impar de consoane)

Obs. E ok și dacă faci verificarea în funcție și returnează 1 sau 0 direct. S-a lasat peste tot la libertatea lor cum împart codul între main și funcție.

0.3p - Afișare secvența de litere dacă respectă condiția (de ex înlocuiesc spațial cu 0, sau copiază în altă parte șirul, sau orice altă metodă)

0.1p - calcul număr total de secvențe cu condiția dată

Afișare în baza 16 (total 0.65p):

0.2p - buclă corectă pt aflarea cifrelor și stocarea lor (pe stivă sau într-un șir)

0.2p - convertirea cifrelor în litere unde e cazul (cu xlat sau manual)

0.15p - printf cu %c (nu le-am dat voie să folosească %x)(Dacă faci printf cu %d fără să verifice că e mai mic ca 10 primesc doar 0.05p)

0.1p - salvare ECX înainte de apel printf pt a putea parcurge "cifrele" în continuare (primesc punctele și dacă nu au folosit ECX, ci altă metodă de iterare)

Modul 2:

0.2p - descriere mecanisme (transmitere procedură (stivă, registrii, variabile))

0.25p - parcurgere secvența până la spațiu sau până la 0

0.25p - identificare consoane și vocale

Seminar 7

Problema 2

Să se scrie un program în limbaj de asamblare care primește la intrare un șir de dublucuvinte definit în segmentul de date. Programul va forma un nou șir prin extragerea octetului superior al cuvântului inferior, urmată de extragerea octetului inferior al cuvântului superior din fiecare dublucuvânt și va forma cu aceștia un șir pe care îl va stoca în memorie. Programul va determina toți octeții care au valori pozitive și pare, sunt diferiți ca valoare și va calcula suma lor. La final, programul va afișa în baza 16 cei k octeți obținuți și suma acestora.

Programul va fi format din două module: modulul „a.asm”, care conține programul principal și modulul „b.asm”, care conține o procedură numită „selectie”. Procedura va primi ca argumente offset-ul șirului de octeți, va determina octeții care îndeplinesc condițiile impuse și va returna suma acestora.

Explicați și detaliați abordarea algoritmului și mecanismele implicate. Justificați și comentați corespunzător textul sursă.

Exemplu:

Șirul de dublucuvinte: 630C018Fh, 640E0563h, 6102DF07h, 6202CF00h, 6506BF02h

Șirul de octeți extrași: 01h, 0Ch, 05h, 0Eh, 0DFh, 02h, 0CFh, 02h, 0BFh, 06h

Șirul de octeți pozitivi: 01h, 0Ch, 05h, 0Eh, 02h, 06h

Șirul de octeți pari și pozitivi: 0Ch, 0Eh, 02h, 06h

Suma octetilor identificati: 22h