Лабораторна робота №10

З системного програмування

Виконав – **Нестерук Юрій, гр. ІО-12**

**Завдання на лабораторну роботу:**

У роботі реалізувати лінійний, бінарний і хеш-пошук. Для перевірки роботи алгоритму використовувати формат таблиць з лаб.раб. 9

**Код програми:**

assume cs: code, ds: data

element struc

key db 255

field db 0

element ends

list struc

key db 255

list ends

;Макрос для виведення результату пошуку

result macro method, res

mov ah, 09h

mov dx, offset method

int 21h

mov dx, offset res

int 21h

endm

;Макрос для отримання хеша ключа

getHashKey macro key, hash

xor ax, ax

xor dx, dx

movzx ax, key

div hash\_size

mov hash, edx

endm

;Макрос для вставки в хеш-таблицю посилання на список

;(елемент структури) за хешем його ключа

insertInTable macro key

getHashKey key, eax

lea edi, hash\_table[eax\*2]

mov [edi], al

;lea eax, key

;mov [edi+1], al

lea ax, key

mov [edi+1], al

endm

data segment use16

elems element <'H', 68h>

element <'G', 67h>

element <'F', 66h>

element <'C', 63h>

element <'D', 64h>

element <'E', 65h>

element <'B', 62h>

endel element <'A', 61h>

first dd ?

last dd ?

num equ 8

find1 db 'E'

find2 db 'N'

findh db 'P'

hash\_size db 8

hash\_table dw 8 dup(?)

lists list <'B'>

list <'G'>

list <'C'>

list <'F'>

list <'D'>

list <'E'>

list <'H'>

list <'A'>

linear db 'Linear$'

linearS db 'Linear (sorted)$'

binary db 'Binary$'

hash db 'Hash$'

success db ' search - success!', 0dh, 0ah, '$'

failure db ' search - failure.', 0dh, 0ah, '$'

data ends

code segment

beg:

.386

mov ax, data

mov ds, ax

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Лінійний пошук

lea ebx, elems

mov ecx, num

xor esi, esi

l1:

mov al, ebx[esi\*2]

cmp al, find1

je s1

inc esi

loop l1

result linear, failure

jmp c1

s1: result linear, success

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

c1: ;Сортування

lea edx, elems

mov ecx, num

dec ecx

li: xor esi, esi

lj: cmp esi, ecx

je lj\_end

mov ax, edx[esi\*2]

mov bx, edx[esi\*2 + 2]

cmp al, bl

jl no\_exch

mov edx[esi\*2], bx

mov edx[esi\*2 + 2], ax

no\_exch:inc esi

jmp lj

lj\_end: loop li

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Лінійний пошук на відсортованому масиві елементів

lea ebx, elems

mov ecx, num

xor esi, esi

l2:

mov al, ebx[esi\*2]

cmp find1, al

je s2

jl f2

inc esi

loop l2

f2: result linearS, failure

jmp c2

s2: result linearS, success

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

c2: ;Бінарний пошук

lea eax, elems

mov first, eax

lea eax, endel

mov last, eax

l3:

mov ebx, first

mov eax, last

sub eax, ebx

shr eax, 1

btr eax, 0

add eax, first

mov dl, [eax]

cmp find1, dl

je s3

jl lk

add eax, 2

mov first, eax

jmp ll

lk: mov last, eax

ll: cmp ebx, last

je f3

jmp l3

mov last, eax

loop l3

s3: result binary, success

jmp c3

f3: result binary, failure

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

c3: ;Додавання елементів (списків) у хеш-таблицю

mov ecx, num

lea ebx, lists

xor esi, esi

l4:

insertInTable lists[esi].key

inc esi

loop l4

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Пошук елемента у хеш-таблиці

getHashKey findh, esi

lea ebx, hash\_table

mov dl, ebx[esi+1]

lea ebx, lists

movzx edi, dl

mov al, ebx[edi]

cmp al, findh

je s4

result hash, failure

jmp c4

s4: result hash, success

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

c4: mov ax, 4c00h

int 21h

code ends

end beg