

Міністерство освіти і науки України
Національний технічний університет України „КПІ”
Факультет інформатики та обчислювальної техніки

Кафедра автоматизованих систем обробки
інформації та управління

ЗВІТ

до лабораторної роботи № 2
з предмету:

„МУЛЬТИПАРАДИГМЕННЕ ПРОГРАМУВАННЯ”

**Виконав
студент**

*ІІІ-61 Кушка Михайло
Олександрович, 3-й курс, ІІІ-6116*

(№ групи, прізвище, ім'я, по батькові, курс, номер
залікової книжки)

Прийняв

Очеретяний О. К.

(посада, прізвище, ім'я, по батькові)

Київ 2018

ЗМІСТ

1. ПОСТАНОВКА ЗАДАЧІ	3
2. ВИСНОВОК.....	4
3. КОД ПРОГРАМИ	6

1. ПОСТАНОВКА ЗАДАЧИ

Задание 1

12.	Описать функцию, которая, выдавала бы атомарный элемент списка по заданному номеру n , считая от начала. Пример : для списка '(2) (3) 4 5 a (e r) g) и $n=3$ результатом будет a.
-----	--

Задание 2

Написать программу сортировки списка методом Шелла. Вычисление последовательности шагов сортировки производится в соответствии с вариантом в Таблице 2.

9 – 16.	Методом, предложенным Дональдом Кнутом : $h_{k-1} = 3 * h_k + 1$, $h_i = 1$, где h_i - шаг сортировки, $i = \lceil \log_3 n \rceil - 1$ - число шагов сортировки, n – длина списка.
---------	---

Задание 3

Написать программу сортировки [6] списка в соответствии с вариантом в таблице 3.

6.	Сортировка Хоара.
----	-------------------

Задание 4

Написать программу объединения двух отсортированных списков в один. При этом порядок сортировки в списке-результате должен сохраняться.

Задание 5

Написать программу в соответствии с заданием из Таблицы 4.

12, 24	Написать программу сортировки списка методом Хоара [6].
--------	---

2. РЕЗУЛЬТАТИ РОБОТИ

A

(4 5 6 7 7 8 9 10 11)

(1 2 3 4 5 6 7 8)

(1 2 3 4 5 6 7 8)

3. ВИСНОВОК

Використав рекурсію для реалізації алгоритмів сортування та більш простих за допомогою використання Lisp. Часом дуже складно реалізувати алгоритм сортування без використання змінних і циклів, а за допомогою лише рекурсії. Причому складність тут не в мові реалізації, а саме в рекурсивному алгоритмі.

4. КОД ПРОГРАМИ

```
; TASK 1
(defun atomicity (lst n)
  (cond ((null lst) nil)
        ((and (atom (car lst)) (eq n '0)) (car lst))
        ((not (atom (car lst)))
         (atomicity (cdr lst) n))
        ((and (atom (car lst)) (not (eq n '0)))
         (atomicity (cdr lst) (- n 1)))))

(print (atomicity '((2) (3) 4 5 a (e r) g) '2))

; TASK 4
(defun mergeo (l1 l2)
  (cond ((null l1) l2)
        ((null l2) l1)
        ((> (car l1) (car l2)) (cons (car l2) (mergeo l1 (cdr l2))))
        (T (cons (car l1) (mergeo (cdr l1) l2)))))

(print (mergeo '(5 6 7) '(4 7 8 9 10 11)))
; (print (mergeo '(7 8 9) '(1 2 3)))

; QuickSort
(defun quicksort (l)
  (cond ((null l) nil)
        (T
         (append
          (quicksort (list< (car l) (cdr l)))
          (list (car l))
          (quicksort (list>= (car l) (cdr l)))))))

(defun list< (a b)
  (cond ((or (null a) (null b)) nil)
        ((< a (car b)) (list< a (cdr b)))
        (T (cons (car b) (list< a (cdr b))))))

(defun list>= (a b)
  (cond ((or (null a) (null b)) nil)
        ((>= a (car b)) (list>= a (cdr b)))
        (T (cons (car b) (list>= a (cdr b))))))

(print (quicksort '(4 2 8 3 1 7 5 6)))

; Shell Sort
(defun insertion (lst x)
  (cond ((null lst) (list x))
        ((> (car lst) x) (cons x lst))
```

```

(T (cons (car lst) (insertion (cdr lst) x))))))

(defun isort (x &optional (s nil))
  (cond ((null x) s)
        (T (isort (cdr x) (insertion s (car x))))))

(defun shello (lst gap finalst)
  (cond ((null lst) finalst)
        (T (cond ((> gap (length lst)) (append finalst (isort lst)))
                  (T (append finalst (isort (subseq lst 0 gap)) (shello (subseq lst gap (length lst))
                                gap finalst))))))

(defun shellSort (lst gaps)
  (cond ((null (cdr gaps)) (shello lst (car gaps) '()))
        (T (shellSort (shello lst (car gaps) '()) (cdr gaps)))))

(defun gaps (len finalLIST)
  '(1 4 13 40 121 364))

(defun shellexecute (lst)
  (shellSort lst (gaps (length lst) '(1))))

(print (shellexecute '(4 2 8 3 1 7 5 6)))
; (print (shellexecute '(3 4 2 1 5 7 8 9 0 6)))

```