Мiнiстерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя Факультет комп'ютерно-інформаційних систем і програмної інженерії

*Кафедра програмної інженері*

ЗВІТ

до лабораторної роботи №2

з навчальної дисципліни  [«Алгоритми і структури даних»](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D1%96%D1%94%D0%BD%D1%82%D0%BE%D0%B2%D0%B0%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)

Тема: «Абстрактний тип даних “Cписок“»

Підготував:

студент групи СП-11

Бондар Віталій Андрійович

Тернопіль 2022

**Мета роботи:**набути навичок з реалізації АТД “Cписок”.

**Теоретичні відомості**

Абстрактний тип даних список LIST можна реалізовувати у вигляді однозвязного та двозвязного списку.

У випадку одновзяного списку, стуктура, яка описує кожен елемент списку має вигляд:

**struct** celltype elementtype element celltype \*next

Для двозвязного списку, в якому містяться два вказівники на попередній та на наступний елементи, стуктура елементу має вигляд:



**struct** celltype elementtype element celltype \*next celltype \*prev

**Завдання 1**. Написати код, реалізації логіки роботи основних операцій однозвязного списку:

* MAKENULL (створює порожній список)
* END (повертає кінець списку)
* FIRST (повертає початок списку)
* INSERT (додає елемент до списку в задану позицію)
* DELETE (видаляє елемент з списку)
* LOCATE (знаходить позицію елементу в списку)
* RETRIEVE (повертає значення елементу списку)
* NEXT (повертає вказівник на наступний елемент списку)

Файлу main.c з проекту містить оголошення функцій для роботи з однозвязним списком, але не містить їхньогї реалізації. Місця, де потрібно вписати код реалізацію алгоритмів, які виконують відповідні операції позначено коментарями /\* НАПИСАТИ КОД \*/ .

При запуску проету на виконання відбуваєтсья автоматична перевірка реалізації функцій при допомозі тестів. Порядок перевірки послідовний. У випадку помилок в реалізції логіки роботи, виводиться повідомлення про перший тест (включає навзу функції), який невдалося пройти, а також повідомлення про причини невдачі. Завдання вважається виконаним, якщо усі тести пройдено.

Лістинг програми

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

#include "linkedlist.h"

LIST MAKENULL(LIST L) {

L = new celltype;

L->next = NONE;

return L;

}

position END(LIST L) {

position pos = L;

while (pos->next != NONE)

pos = pos->next;

return pos;

}

position FIRST(LIST L) {

return L->next;

}

position NEXT(position p, LIST L) {

return p->next;

}

LIST INSERT(elementtype x, position p, LIST L) {

position pos = p->next;

p->next = newnode();

p->next->element = x;

p->next->next = pos;

return NONE;

}

LIST DELETE(position p, LIST L) {

if (p->next != NONE) {

p->next = p->next->next;

}

return L;

}

elementtype RETRIEVE(position p, LIST L) {

return p->next->element;

}

position LOCATE(elementtype x, LIST L) {

position pos = L;

while (pos->next != NONE)

{

if (pos->next->element == x)return pos;

pos = pos->next;

}

return NONE;

}

int main(int argc, char\*\* argv) {

run();

system("PAUSE");

return 0;

}

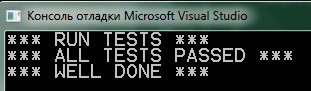


Рисунок 1 – Результат виконання завдання 1

**Завдання 2**. Написати код, реалізації логіки роботи основних операцій двозвязного списку:

* MAKENULL (створює порожній список)
* END (повертає кінець списку)
* FIRST (повертає початок списку)
* INSERT (додає елемент до списку в задану позицію)
* DELETE (видаляє елемент з списку)
* LOCATE (знаходить позицію елементу в списку)
* RETRIEVE (повертає значення елементу списку)
* PREVIOUS (повертає вказівник на попередній елемент списку)
* NEXT (повертає вказівник на наступний елемент списку)

Файлу main.c з проекту містить оголошення функцій для роботи з двозязним списком, але не містить їхньогї реалізації. Місця, де потрібно вписати код реалізацію алгоритмів, які виконують відповідні операції позначено коментарями /\* НАПИСАТИ КОД \*/ .

Лістинг програми

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

using namespace std;

typedef char elementtype;

typedef struct A {

elementtype element;

A\* prev;

A\* next;

} celltype;

typedef celltype\* LIST; typedef celltype\* position;

#define newnode() (celltype\*)malloc(sizeof(celltype))

#define NONE (celltype\*)NULL

LIST MAKENULL(LIST L) {

L = newnode();

L->next = NONE;

L->prev = NONE;

return L;

}

position END(LIST L) {

position pos = L;

while (pos->next != NONE) {

pos = pos->next;

}

return pos;

}

position FIRST(LIST L) {

return L;

}

LIST INSERT(elementtype x, position p, LIST L) {

position pos = p->next;

p->next = newnode();

p->next->next = pos;

p->next->prev = p;

p->next->element = x;

return L;

}

LIST DELETE(position p, LIST L) {

if (p->next != NONE)

{

p->next = p->next->next;

if (p->next != NONE) {

p->next->prev = p;

}

}

return L;

}

position LOCATE(elementtype x, LIST L) {

position pos = L;

while (pos->next != NONE)

{

if (pos->next->element == x) {

return pos;

pos = pos->next;

}

}

return NONE;

}

elementtype RETRIEVE(position p, LIST L) {

return p->next->element;

}

position PREVIOUS(position p, LIST L) {

return p->prev;

}

position NEXT(position p, LIST L) {

return p->next;

}

int main(int argc, char\*\* argv) {

LIST L=MAKENULL(newnode());

char mas[33]{ 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R',

'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z' };

for (int i = 0; i < 33; i++)

{

INSERT(mas[i], END(L), L);

}

PRINTLIST(L);

return 0;

}

**Завдання 3**. З використанням операцій АТД Список написати реалізацію функції PRINTLIST, яка друкує перелік усіх елементі, що є у списку.

void PRINTLIST(LIST L)

{

for (position p = FIRST(L); p->next != (END(L))->next; p = NEXT(p, L))

{

cout << RETRIEVE(p, L);

cout << " ";

}

}

int main(int argc, char\*\* argv) {

LIST L=MAKENULL(newnode());

char mas[33]{ 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R',

'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z' };

for (int i = 0; i < 33; i++)

{

INSERT(mas[i], END(L), L);

}

PRINTLIST(L);

return 0;

}



Рисунок 2 - Результат виконанян завдання 3

Висновок: набули навичок з реалізації АТД “Cписок”.