

Тема: Динамічні списки.

Мета: Навчитись працювати з динамічними списками

Індивідуальне завдання

Завдання на оцінку “задовільно”:

На базі попередньо розробленого функціоналу по роботі з прикладною областю сформуванню односпрямований список елементів розробленої структури. Реалізувати наступні функції роботи зі списком:

- читання даних з файлу, використовуючи функцію `fscanf`;
- запис даних у файл, використовуючи функцію `fprintf`;
- вивід вмісту списку на екран;
- пошук у списку об'єкта за заданим критерієм;
- додавання об'єкта у кінець списку;
- видалення об'єкта зі списку.

Завдання на оцінку “добре”:

- виконати завдання на “задовільно”;
- реалізувати діалоговий режим спілкування з користувачем за допомогою меню;
- сортування вмісту списку за одним з критеріїв. При цьому дозволяється міняти місцями не вказівники на об'єкти, а вміст об'єктів (за допомогою третьої змінної);
- переробити метод додавання з можливістю вставлення додаткового елемента після будь-якого елемента списку;
- продемонструвати відсутність витоків пам'яті;
- розробити модульні тести, що демонструють коректність роботи реалізованих функцій
- проект має складатися мінімум з 6 файлів (`main.c`, `test.c`, `list.h`, `list.c`, `data.c`, `data.h`)

Завдання на оцінку “відмінно”:

- виконати завдання на “добре”, але замість односпрямованого списку треба використовувати двоспрямований;
- Реалізувати сортування вмісту списку за одним з критеріїв. При цьому обов'язково забезпечити, щоб обмін місцями об'єктів здійснювався шляхом обміну їх покажчиків.

Хід роботи

```

node_t* read_from_file() {
    FILE* f = fopen(filename, "r");

    struct Message message;
    node_t* head = NULL;

    head = (node_t*)malloc(sizeof(node_t));
    head->message = create(0);
    head->next = NULL;

    for(int i=0;i<3;i++){
        fgets(message.topic,64,f);
        fgets(message.text, 256, f);
        fscanf(f, "%s%s%s%s%d\n", message.sender.name, message.sender.mail,
            message.recipient.name, message.recipient.mail, &message.encoding, &message.draft);
        print(message);
        add(head, message);
    }
    remove_first(&head);
    fclose(f);
    return head;
}

```

Рис.1 - читання даних з файлу

Your choice: 6

From:exampleSender (exampleMail2@gmail.com)
To:exampleRecipient (exampleMail2@gmail.com)
Topic: exampleTopic

Text:
example Text for this cal

Encoding:UTF-16

From:exampleSender (exampleMail1@gmail.com)
To:exampleRecipient (exampleMail1@gmail.com)
Topic: exampleTopic

Text:
exampleText

Encoding:CP-1251
[Draft]

From:exampleSender (exampleMail3@gmail.com)
To:exampleRecipient (exampleMail3@gmail.com)
Topic: exampleTopic

Text:
exampleText

Encoding:CP-1251

Successful.

Результат зчитування з файлу.(Дивіться вміст файлу нижче)

```
void write_to_file(node_t* head) {  
    node_t* current = head;  
    FILE* file;  
    file = fopen(filename, "w");  
    fprintf(file, "");  
    fclose(file);  
    while (current != NULL) {  
        fwrite(current->message);  
        current = current->next;  
    }  
}
```

Рис.2 - запис даних у файл

```
From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]
```

```
From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16
```

```
From:exampleSender (exampleMail1@gmail.com)
To:exampleRecipient (exampleMail1@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
```

Successful.

Зміст списку у консолі.

```
exampleTopic
exampleText
exampleSender
exampleMail0@gmail.com
exampleRecipient
exampleMail0@gmail.com
2
1
exampleTopic
exampleText
exampleSender
exampleMail0@gmail.com
exampleRecipient
exampleMail0@gmail.com
1
0
exampleTopic
exampleText
exampleSender
exampleMail1@gmail.com
exampleRecipient
exampleMail1@gmail.com
2
0
exampleTopic
exampleText
exampleSender
exampleMail2@gmail.com
exampleRecipient
exampleMail2@gmail.com
0
0
exampleTopic
exampleText
exampleSender
exampleMail3@gmail.com
exampleRecipient
exampleMail3@gmail.com
1
0
```

Вміст файлу після запису у файл.

```

void print_list(node_t* head) {
    node_t* current = head;

    while (current != NULL) {
        print(current->employer);
        current = current->next;
    }
}

```

Рис.3 - вивід вмісту списку.

```

A list of Messages.Select an action from the following.

1.Write list to screen.
2.Generate and add message.
3.Remove message by index.
4.Search draft messages.
5.Sort by encoding.
6.Reading messages from a file.
7.Writing messages in file.
0.Exit.

Your choice: 1

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]

Successful.

```

Результат роботи функції виводу.

```

void search_draft(node_t* head) {
    node_t* current = head;

    while (current->next != NULL) {
        if (current->message.draft == true)
            print(current->message);
        current = current->next;
    }
}

```

Рис.4 - пошук по заданому критерію.

Your choice: 4

```
From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]
```

```
From:exampleSender (exampleMail4@gmail.com)
To:exampleRecipient (exampleMail4@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]
```

Successful.

Результат роботи пошуку по заданому критерію страхування.

```
void add(node_t* head, Message message) {
    node_t* current = head;

    while (current->next != NULL) {
        current = current->next;
    }

    current->next = (node_t*)malloc(sizeof(node_t));
    current->next->message = message;
    current->next->next = NULL;
}
```

Рис.5 - додавання об'єкту у кінець списку

Your choice: 2

Successful.

A list of Messages.Select an action from the following.

- 1.Write list to screen.
- 2.Generate and add message.
- 3.Remove message by index.
- 4.Search draft messages.
- 5.Sort by encoding.
- 6.Reading messages from a file.
- 7.Writing messages in file.
- 0.Exit.

Your choice: 1

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

Successful.

Результат створення та додавання в кінець списку нового повідомлення.


```
void remove_by_index(node_t** head, int n) {  
  
    int i = 0;  
  
    node_t* current = *head;  
    node_t* temp_node = NULL;  
  
    if (n == 0) {  
        pop(head);  
        return;  
    }  
  
    for (int i = 0; i < n - 1; i++) {  
        if (current->next == NULL) {  
            return;  
        }  
        current = current->next;  
    }  
  
    temp_node = current->next;  
    current->next = temp_node->next;  
    free(temp_node);  
  
    return;  
}
```

Рис. 6 - видалення об'єкту зі списку.

Your choice: 3

Choose index(0-2):0

Successful.

A list of Messages.Select an action from the following.

- 1.Write list to screen.
- 2.Generate and add message.
- 3.Remove message by index.
- 4.Search draft messages.
- 5.Sort by encoding.
- 6.Reading messages from a file.
- 7.Writing messages in file.
- 0.Exit.

Your choice: 1

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

From:exampleSender (exampleMail1@gmail.com)
To:exampleRecipient (exampleMail1@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251

Successful.

Результат видалення повідомлення зі списку за індексом.

```

void menu(node_t* head) {
    printf("\n\n      A list of Messages.Select an action from the following.\n\n");
    printf("1.Write list to screen.\n");
    printf("2.Generate and add message.\n");
    printf("3.Remove message by index.\n");
    printf("4.Search draft messages.\n");
    printf("5.Sort by encoding.\n");
    printf("6.Reading messages from a file.\n");
    printf("7.Writing messages in file.\n");
    printf("0.Exit.\n\n");
    printf("Your choice: ");

    int choice = -1;
    int size_list;

    while (choice < 0 || choice>7) {
        scanf("%d", &choice);
        if (choice < 0 || choice>7)
            printf("You entered an invalid value, please retry\n\nYour choice: ");
    }
    printf("\n\n");

    switch (choice) {
    case 0: return;
    case 1: print_list(head);
            break;
    case 2: add(head, create(size(head)));
            break;
    case 3:
        size_list = size(head);
        choice = -1;
        printf("Choose index(0-%d):", size_list);
        while (choice < 0 || choice > size_list) {
            scanf("%d", &choice);
            if (choice < 0 || choice > size_list)
                printf("You entered an index that is larger than the size of the list, please retry\n\nYour choice: ");
        }
        remove_by_index(&head, choice);
        break;
    case 4: search_draft(head);
            break;
    case 5: head = sort_by_encoding(head);
            break;
    case 6: head=read_from_file();
            break;
    case 7: write_to_file(head);
            break;
    }
    printf("Successful.");
    menu(head);
}

```

Рис.7-8 Діалогове меню.

```

node_t* sort_by_encoding(node_t* head) {
    node_t* tmp = NULL;

    while (head != NULL) {
        node_t* node = head;
        head = head->next;

        if (tmp == NULL || node->message.encoding < tmp->message.encoding)
        {
            node->next = tmp;
            tmp = node;
        }
        else
        {
            node_t* current = tmp;
            while (current->next != NULL && !(node->message.encoding < current->next->message.encoding))
            {
                current = current->next;
            }

            node->next = current->next;
            current->next = node;
        }
    }
    return tmp;
}

```

Рис.9 - сортування за критерієм.

Your choice: 1

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

From:exampleSender (exampleMail1@gmail.com)
To:exampleRecipient (exampleMail1@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251

From:exampleSender (exampleMail2@gmail.com)
To:exampleRecipient (exampleMail2@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-8

From:exampleSender (exampleMail3@gmail.com)
To:exampleRecipient (exampleMail3@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

Successful.

Список до сортування.

Your choice: 1

From:exampleSender (exampleMail2@gmail.com)
To:exampleRecipient (exampleMail2@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-8

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

From:exampleSender (exampleMail3@gmail.com)
To:exampleRecipient (exampleMail3@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:UTF-16

From:exampleSender (exampleMail0@gmail.com)
To:exampleRecipient (exampleMail0@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251
[Draft]

From:exampleSender (exampleMail1@gmail.com)
To:exampleRecipient (exampleMail1@gmail.com)
Topic: exampleTopic
Text:
exampleText
Encoding:CP-1251

Список після сортування.

```
void add_to_beginning(node_t** head, Message employer) {  
  
    node_t* new_node;  
    new_node = (node_t*)malloc(sizeof(node_t));  
  
    new_node->message = employer;  
    new_node->next = *head;  
    *head = new_node;  
}
```

Рис.10 - додавання з можливістю встановлення додаткового елементу.

Висновок: в лабораторній роботі отримані навички з роботою динамічних списків, а також зчитування за допомогою fscanf, запис fprintf, вивід вмісту списку на екран, пошук за заданим критерієм та додавання у кінець списку, видалення об'єкта зі списку