

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«Програмування ч.2»

*Звіт з лабораторної роботи №18-19*  
*Тема: «Динамічні масиви та динамічні списки»*

Виконав:  
ст. гр. КІТ-120А  
Старовойтов Н.А.

Перевірив:  
Челак В.В.

Харків – 2021

**Мета:** Отримати навички роботи із динамічними масивами та динамічними списками.

### ***Індивідуальне завдання***

Робота на оцінку “відмінно”.

Лабораторна робота №18:

1. Розробити функцію, яка реалізує вставку в рядок “s” другий рядок “s2” в “i”-у позицію рядка “s”.
2. Розробити функцію видалення з рядка “s” усіх символів з індексами в заданому діапазоні.
3. За допомогою функцій memscr, memset створити функції додання та видалення елементів з динамічного масиву вашої прикладної області.
4. Додати модульні тести, що демонструють коректність розроблених функцій

Лабораторна робота №19:

1. Розробити функцію, яка зчитує список з файлу
2. Розробити функцію, яка записує дані(список) у файл;
3. Розробити функцію, яка виводить список на екран;
4. Розробити функцію, яка буде сортувати список за заданим критерієм;
5. Розробити функцію, яка буде знаходити заданий елемент у списку;
6. Розробити функцію, яка видаляє елемент зі списку;
7. Розробити функцію, яка вставляє новий елемент по індексу;
8. Реалізувати діалоговий режим спілкування з користувачем

## Опис програми

### Поля даних

int	LOTR
char	name [15]
int	age
struct Bird::Home	home
enum Sex	enumSex
struct Bird *	next
struct Bird *	prev

Рисунок 1 — Структура Bird

### Функції

void	<b>read_from_file</b> (struct Bird *birds)	Функція читає данні з заданого файлу та записує їх в задану структуру Детальніше...
void	<b>random_age_generation</b> (struct Bird *birds)	Функція генерує випадковий вік птиці у заданому діапазоні Детальніше...
void	<b>print_in_console</b> (struct Bird *birds, int count)	Функція виводить масив структур на екран Детальніше...
int	<b>bool_compare_strings</b> (char a[10], char b[10])	Функція порівнює строки між собою та повертає результат у форматі, що легко привести до типу bool. Детальніше...
struct Bird *	<b>add_bird</b> (struct Bird *birds, struct Bird paste_bird, int pos)	Функція вставляє елемент в масив структур за заданим індексом Детальніше...
struct Bird *	<b>delete_bird</b> (struct Bird *birds, int pos)	Функція видаляє елемент з масиву структур за заданим індексом Детальніше...
char *	<b>insert_string</b> (char *text, char *pasted_text, int pos)	Функція вставляє підрядок в рядок за заданим індексом Детальніше...
char *	<b>delete_string</b> (char *text, int pos1, int pos2)	Функція видаляє підрядок з тексту за заданим індексом Детальніше...
void	<b>Demonstration_of_strings</b> ()	Функція демонструє реалізацію перших двох завдань: робота з рядками Детальніше...

Рисунок 2 — Всі функції лабораторної роботи №18

## Функції

<code>struct List *</code>	<code>New_list ()</code>	Функція створює список Детальніше...
<code>struct Bird *</code>	<code>New_bird (struct Bird *bird)</code>	Функція створює новий елемент списку Детальніше...
<code>void</code>	<code>Add_element_to_end (struct List *list, struct Bird *bird)</code>	Функція додає новий елемент в кінець списку Детальніше...
<code>void</code>	<code>Add_element_to_begin (struct List *list, struct Bird *bird)</code>	Функція додає новий елемент в початок списку Детальніше...
<code>void</code>	<code>Insert_element (struct List *list, int pos, struct Bird *bird)</code>	Функція додає новий елемент в список по індексу Детальніше...
<code>void</code>	<code>Delete_element (struct List *list, int pos)</code>	Функція видаляє елемент зі списку Детальніше...
<code>void</code>	<code>Read_list_from_file (struct List *list)</code>	Функція читає данні з заданого файлу та записує їх в список Детальніше...
<code>void</code>	<code>Show_list (struct List *list)</code>	Функція виводить список на екран Детальніше...
<code>void</code>	<code>Print_list_in_file (struct List *list)</code>	Функція записує у файл заданий список Детальніше...
<code>void</code>	<code>Find_bird (struct List *list, int index)</code>	Функція знаходить заданий елемент списку Детальніше...
<code>void</code>	<code>Sort_birds_by_age (struct List *list)</code>	Функція сортує список за віком птахів Детальніше...
<code>void</code>	<code>Free_list (struct List *list)</code>	Функція очищує список Детальніше...

Рисунок 3 — Всі функції лабораторної роботи №19

## Схеми алгоритмів функцій

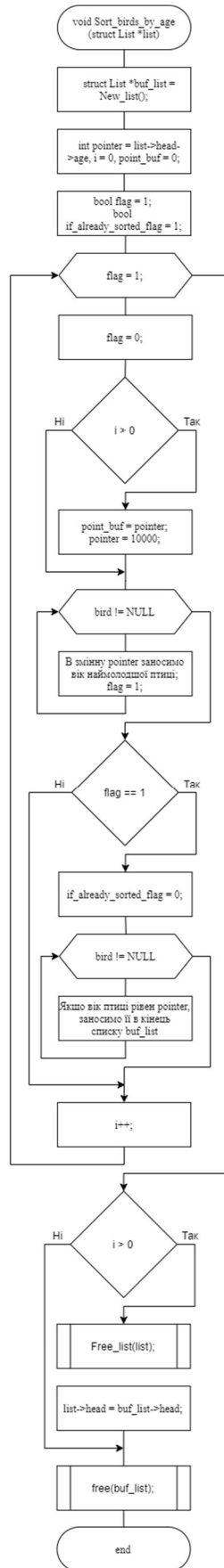


Рисунок 4 — Блок-схема функції Sort\_birds\_by\_age

## Текст програми

### Лабораторна робота №18:

```
#include "lib.h"

int main() {

    Demonstration_of_strings();

    struct Bird *birds = malloc(N * sizeof(struct Bird));
    read_from_file(birds);
    random_age_generation(birds);

    print_in_console(birds, N);

    struct Bird paste_bird = {1, "Some_bird", 27, 48, 49, 2, 0, Male};

    int position, a;
    printf("Select add/remove (1/0): ");
    scanf("%d", &a);
    if (a == 1) {
        printf("Enter a position to paste: ");
        scanf("%d", &position);
        birds = add_bird(birds, paste_bird, position);

printf("\n\n_____
\n\n");
        print_in_console(birds, N + 1);
    }
    if (a == 0) {
        printf("Enter a position to remove: ");
        scanf("%d", &position);
        birds = delete_bird(birds, position);

printf("\n\n_____
\n\n");
        print_in_console(birds, N - 1);
    }

    free(birds);
    return 0;
}
```

### Лабораторна робота №19:

```
#include "list.h"
#include "data.h"

int main() {
    system("chcp 65001");
    printf("\n");
    struct List *list = New_list();

    struct Bird struct_paste_bird = {0, "SOME_BIRD", 27, 250, 40, 2, 0, 1};
    struct Bird *paste_bird = malloc(sizeof(struct Bird));
```

```

*paste_bird = struct_paste_bird;

Read_list_from_file(list);

while (true) {
    int action;
    printf("\n|-----|\n");
    printf(" Выберите действие: \n");
    printf("\t1. Вывести список на экран\n");
    printf("\t2. Вывести список в файл\n");
    printf("\t3. Вставить новый элемент в список\n");
    printf("\t4. Удалить элемент из списка\n");
    printf("\t5. Найти в списке элемент\n");
    printf("\t6. Сортировать список по возрасту птицы\n");
    printf("\t7. Выход\n");
    printf("|-----|\n\n");
    printf(" --> ");
    scanf("%d", &action);
    switch (action) {
        case 1:
            printf("\nСписок:\n\n");
            Show_list(list);
            break;
        case 2:
            Print_list_in_file(list);
            printf("\n Список занесен в файл 'result.txt'\n\n");
            break;
        case 3:
            printf("\nЭлемент, который будет вставлен: \n\n");
            Print_bird(paste_bird);
            printf("\nВведите позицию, на которую нужно вставить элемент: \n");
            printf(" --> ");
            int pos;
            scanf("%d", &pos);
            Insert_element(list, pos, paste_bird);
            printf("\nЭлемент вставлен\n\n");
            break;
        case 4:
            printf("\nВведите позицию элемента, который будет удалён: \n");
            printf(" --> ");
            int num;
            scanf("%d", &num);
            Delete_element(list, num);
            printf("\nЭлемент удалён\n\n");
            break;
        case 5:
            printf("\nВведите критерий, по которому искать птицу: \n");
            printf("\t1. С кольцом или без\n");
            printf("\t2. Имя\n");
            printf("\t3. Возраст\n");
            printf("\t4. Площадь скворечника\n");
            printf("\t5. Высота скворечника\n");
            printf("\t6. Кол-во кормушек скворечника\n");
            printf("\t7. Является ли скворечник гнездом\n");
            printf("\t8. Пол птицы\n");
            printf(" --> ");
            int number;
            scanf("%d", &number);
            printf("\n");
            Find_bird(list, number);
            break;
        case 6:

```

```

        Sort_birds_by_age(list);
        printf("\nСписок отсортирован \n");
        break;
    case 7:
        Free_list(list);
        free(list);
        printf("\nЗавершение работы программы...\n");
        return 0;
    default:
        printf("\nERROR: Вы ввели некорректный вариант\n");
        break;
}
}
}

```

## Результати роботи програми

The screenshot shows a terminal window with the following output:

```

C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab18\src\cmake-build-debug\src.exe
Input text:aaaaaa
Input string you want to insert:FFF
Position:3
Result: aaaFFfaaa

Input text:AAA___AAA
First position to delete:3
Second position to delete:5
Result: AAAAAA

Для продолжения нажмите любую клавишу . . .

```

Рисунок 5 — Результат виконання перших двох завдань лабораторної роботи №18

The screenshot shows a terminal window with the following output:

```

Sex: female

Is bird the Lord of the Ring: NO
Name: Parrot
Age: 148 months
Home area: 2400 sm^2
Home height: 100 sm
Home feeders amount: 1
Is bird's home a nest: NO
Sex: male

Select add/remove (1/0):0
0
Enter a position to remove:1
1

-----

Is bird the Lord of the Ring: Yes
Name: Grus
Age: 64 months
Home area: 2500 sm^2
Home height: 30 sm
Home feeders amount: 0
Is bird's home a nest: Yes

```

Рисунок 6 — Результат виконання третього завдання лабораторної роботи №18





The screenshot shows a terminal window titled 'Run: test'. The command executed is 'C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab18\test\cmake-build-debug\test.exe'. The output displays four test cases, all of which passed: 'Testing add\_bird function:', 'Testing delete\_bird function:', 'Testing insert\_string function:', and 'Testing delete\_string function:'. The final message is 'Process finished with exit code 0'.

```
Run: test x
C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab18\test\cmake-build-debug\test.exe
Testing add_bird function:
Test passed!

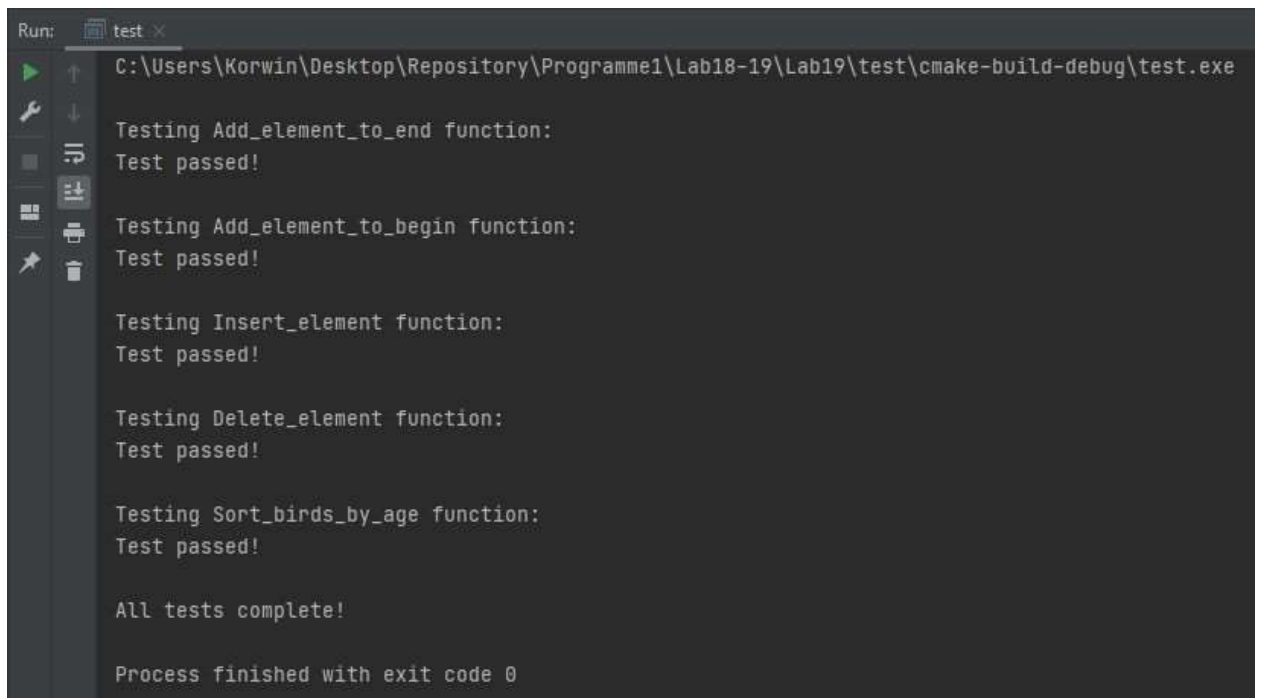
Testing delete_bird function:
Test passed!

Testing insert_string function:
Test passed!

Testing delete_string function:
Test passed!

Process finished with exit code 0
```

Рисунок 7 — Результат успішного виконання тестів лабораторної роботи №18



The screenshot shows a terminal window titled 'Run: test'. The command executed is 'C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab19\test\cmake-build-debug\test.exe'. The output displays five test cases, all of which passed: 'Testing Add\_element\_to\_end function:', 'Testing Add\_element\_to\_begin function:', 'Testing Insert\_element function:', 'Testing Delete\_element function:', and 'Testing Sort\_birds\_by\_age function:'. The final message is 'All tests complete!' followed by 'Process finished with exit code 0'.

```
Run: test x
C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab19\test\cmake-build-debug\test.exe
Testing Add_element_to_end function:
Test passed!

Testing Add_element_to_begin function:
Test passed!

Testing Insert_element function:
Test passed!

Testing Delete_element function:
Test passed!

Testing Sort_birds_by_age function:
Test passed!

All tests complete!

Process finished with exit code 0
```

Рисунок 8 — Результат успішного виконання тестів лабораторної роботи №19

```
Run: src
C:\Users\Korwin\Desktop\Repository\Programme1\Lab18-19\Lab19\src\cmake-build-debug\src.
Active code page: 65001

|-----|
Выберите действие:
1. Вывести список на экран
2. Вывести список в файл
3. Вставить новый элемент в список
4. Удалить элемент из списка
5. Найти в списке элемент
6. Сортировать список по возрасту птицы
7. Выход
|-----|

-->4
4

Введите позицию элемента, который будет удалён:
-->1
1

Элемент удалён

|-----|
Выберите действие:
1. Вывести список на экран
2. Вывести список в файл
3. Вставить новый элемент в список
4. Удалить элемент из списка
5. Найти в списке элемент
6. Сортировать список по возрасту птицы
7. Выход
|-----|

-->6
6

Список отсортирован

|-----|
Выберите действие:
1. Вывести список на экран
2. Вывести список в файл
3. Вставить новый элемент в список
4. Удалить элемент из списка
5. Найти в списке элемент
6. Сортировать список по возрасту птицы
7. Выход
|-----|

-->7
7

Завершение работы программы...
Process finished with exit code 0
```

Рисунок 9 — Результат успішного виконання програми лабораторної роботи №19

## **Висновки**

Під час виконання даної лабораторної роботи було отримано навички роботи із динамічними масивами та динамічними списками.