

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«Програмування ч.2»

Звіт з лабораторної роботи №20
Тема: «Макровизначення»

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

Перевірив:
Челак В.В.

Харків – 2021

Мета: Отримати навички роботи із макровизначеннями.

Індивідуальне завдання

Робота на оцінку “відмінно”.

Для попередньо розробленого функціоналу по роботі з прикладною областю за допомогою макровизначень зробити наступну “кастомізацію”: якщо визначене макровизначення `debug`, то при виклику кожної функції роботи з прикладною областю виконувати таке:

- виводити поточну дату та час;
- виводити ім’я поточної функції;
- при завершенні програми вивести загальний час роботи програми.

Також змінити модульні тести таким чином, щоб:

- при запуску кожного тесту виводити ім’я тесту, а по завершенні тесту виводити час, що було витрачено на виконання даного тесту;
- по завершення всіх тестів було виведено час їх виконання.

Опис програми

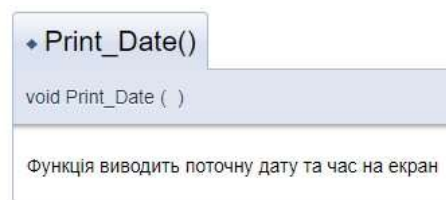


Рисунок 1 — Опис функції `Print_Date`

Функція `Print_Date`:

```
void Print_Date(){
    struct tm *u;
    time_t timer = time(NULL);
    u = localtime(&timer);
    char s[60];
    for (int i = 0; i < 60; i++) s[i] = 0;
    switch (u->tm_wday)
    {
        case 0: strcpy(s, " Воскресенье, "); break;
```

```

case 1: strcpy(s, " Понедельник, "); break;
case 2: strcpy(s, " Вторник, "); break;
case 3: strcpy(s, " Среда, "); break;
case 4: strcpy(s, " Четверг, "); break;
case 5: strcpy(s, " Пятница, "); break;
case 6: strcpy(s, " Суббота, "); break;
}
char tmp[20];
strftime(tmp, 40, "%d.%m.%Y, %H:%M:%S.", u);
strcat(s, tmp);
printf("\nТекущие дата и время:%s", s);
}

```

Схеми алгоритмів функцій

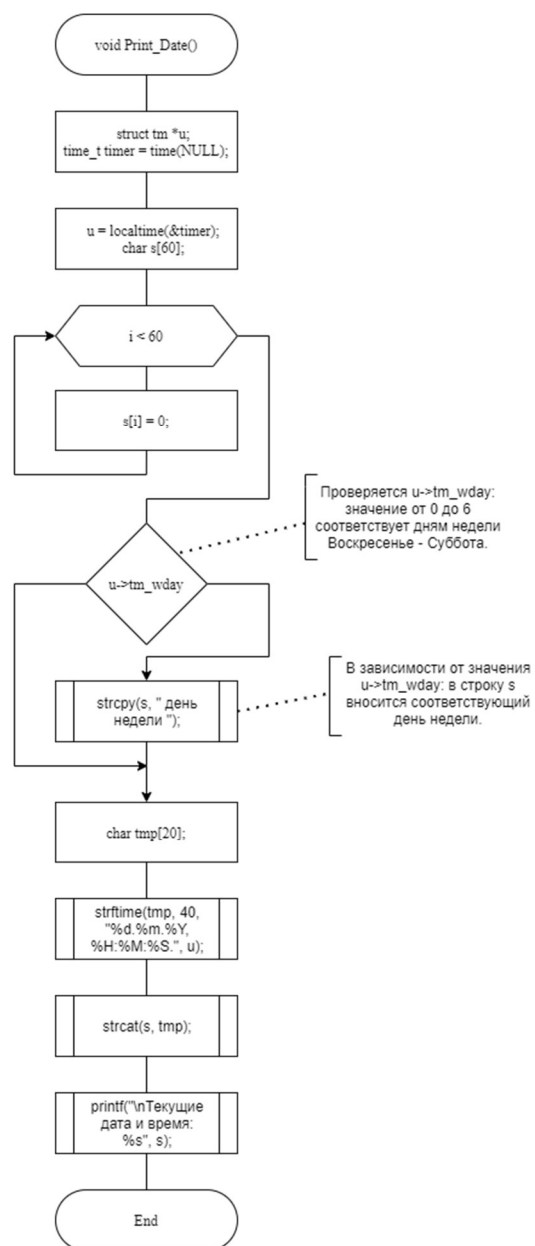


Рисунок 2 — Блок-схема функції Print_Date

Текст программы

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

int main() {
    system("chcp 65001");
    #define DEBUG
    #ifdef DEBUG
        Print_Date();
    #endif
    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);
    int start1, finish1;
    while (true) {
        int action = 0;
        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");
        printf(" --> ");
        scanf("%d", &action);
        switch (action) {
            case 1:
                #ifdef DEBUG
                    printf("\nВыполняется функция Show_list");
                    printf("\nСписок:\n\n");
                    start1 = clock();
                    Show_list(list);
                    finish1 = clock();
                    finish1 -= start1;
                    printf("Время выполнения функции: %dms%s", finish1, "\n");
                #else
                    printf("\nСписок:\n\n");
                    Show_list(list);
                #endif
                break;
            case 2:
                #ifdef DEBUG
                    printf("\nВыполняется функция Print_list_in_file");
                    start1 = clock();
                    Print_list_in_file(list);
                    finish1 = clock();
                    finish1 -= start1;
                    printf("\n Список занесен в файл 'result.txt'\n");
                    printf("Время выполнения функции: %dms%s", finish1, "\n\n");
                #else
                    Print_list_in_file(list);
                    printf("\n Список занесен в файл 'result.txt'\n\n");
                #endif
                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);
    #ifdef DEBUG
    printf("\nВыполняется функция Insert_element");
    start1 = clock();
    Insert_element(list, pos, paste_bird);
    finish1 = clock();
    finish1 -= start1;
    printf("\nЭлемент вставлен\n");
    printf("Время выполнения функции: %dms%s", finish1, "\n\n\n");
    #else
    Insert_element(list, pos, paste_bird);
    printf("\nЭлемент вставлен\n\n\n");
    #endif
    break;

case 4:
    printf("\nВведите позицию элемента, который будет удалён: \n");
    printf("--> ");
    int num;
    scanf("%d", &num);
    #ifdef DEBUG
    printf("\nВыполняется функция Delete_element");
    start1 = clock();
    Delete_element(list, num);
    finish1 = clock();
    finish1 -= start1;
    printf("\nЭлемент удалён\n");
    printf("Время выполнения функции: %dms%s", finish1, "\n\n\n");
    #else
    Delete_element(list, num);
    printf("\nЭлемент удалён\n\n\n");
    #endif
    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");
    #ifdef DEBUG
    printf("\nВыполняется функция Find_bird");
    start1 = clock();
    Find_bird(list, number);
    finish1 = clock();
    finish1 -= start1;
    printf("Время выполнения функции: %dms%s", finish1, "\n\n\n");
    #else
    Find_bird(list, number);
    #endif

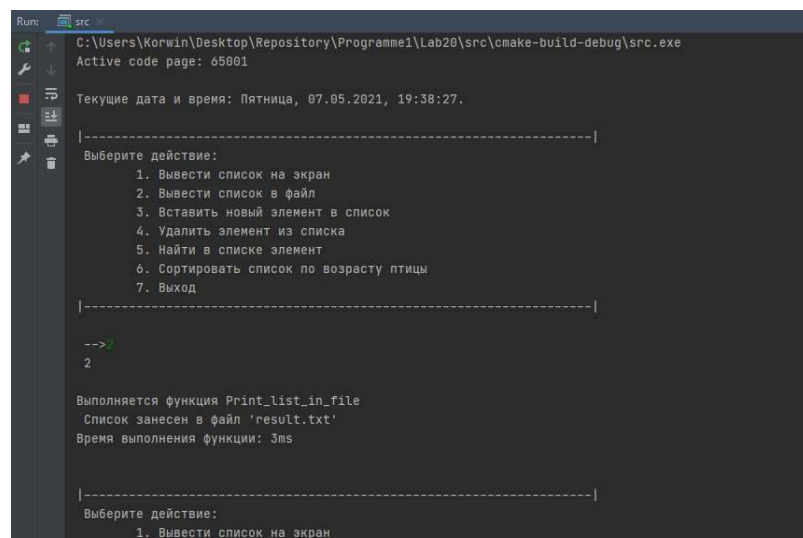
```

```

        break;
    case 6:
        #ifdef DEBUG
        printf("\nВыполняется функция Sort_birds_by_age");
        start1 = clock();
        Sort_birds_by_age(list);
        finish1 = clock();
        finish1 -= start1;
        printf("\nСписок отсортирован \n");
        printf("Время выполнения функции: %dms%s", finish1, "\n\n\n");
        #else
        Sort_birds_by_age(list);
        printf("\nСписок отсортирован \n");
        #endif
        break;
    case 7:
        #ifdef DEBUG
        printf("\nВыполняется функция Free_list");
        start1 = clock();
        Free_list(list);
        finish1 = clock();
        finish1 -= start1;
        free(list);
        printf("\nВремя выполнения функции: %dms%s", finish1, "\n");
        printf("\nЗавершение работы программы...\n");
        return 0;
        #else
        Free_list(list);
        free(list);
        printf("\nЗавершение работы программы...\n");
        return 0;
        #endif
    default:
        printf("\nERROR: Вы ввели некорректный вариант\n");
        break;
}
}
}

```

Результати роботи програми



```

Run: src
C:\Users\Korwin\Desktop\Repository\Programme1\Lab20\src\cmake-build-debug\src.exe
Active code page: 65001

Текущие дата и время: Пятница, 07.05.2021, 19:38:27.

-----|
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
-----|

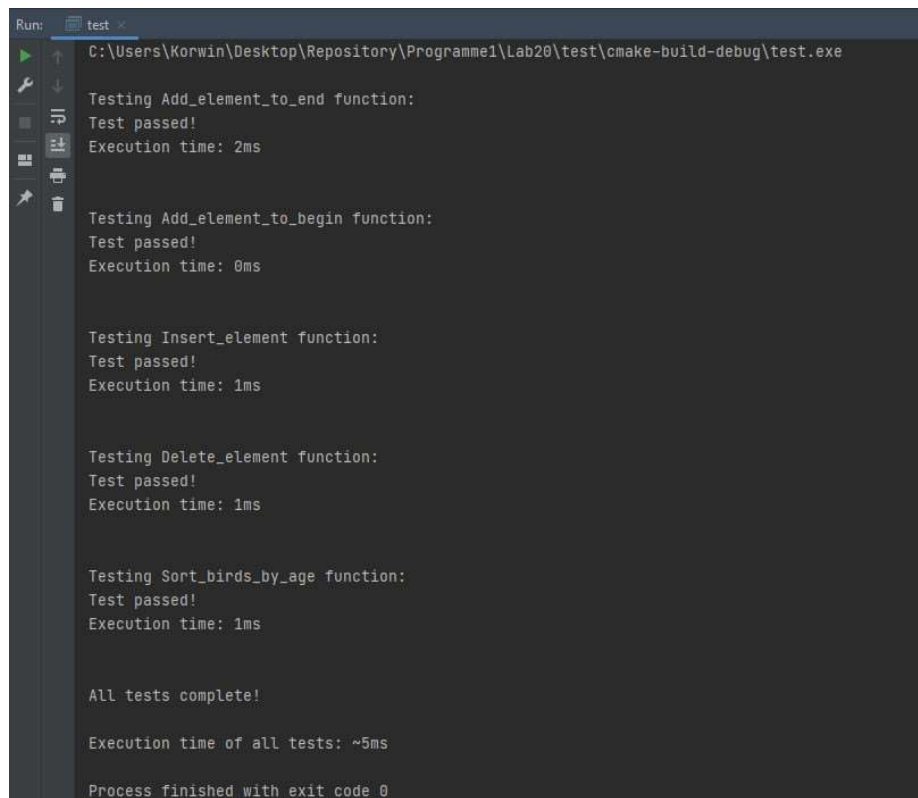
--> 2

Выполняется функция Print_list_in_file
Список занесен в файл 'result.txt'
Время выполнения функции: 3ms

-----|
Выберите действие:
1. Вывести список на экран

```

Рисунок 3 — Результат успішного виконання програми



```
Run: test x
C:\Users\Korwin\Desktop\Repository\Programme1\Lab20\test\cmake-build-debug\test.exe

Testing Add_element_to_end function:
Test passed!
Execution time: 2ms

Testing Add_element_to_begin function:
Test passed!
Execution time: 0ms

Testing Insert_element function:
Test passed!
Execution time: 1ms

Testing Delete_element function:
Test passed!
Execution time: 1ms

Testing Sort_birds_by_age function:
Test passed!
Execution time: 1ms

All tests complete!

Execution time of all tests: ~5ms

Process finished with exit code 0
```

Рисунок 4 — Результат успішного виконання тестів

Висновки

Під час виконання даної лабораторної роботи було отримано навички роботи із макровизначеннями.