



МИНИСТЕРСТВО НАУКИ
И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»



**НГТУ
НЭТИ**

**Факультет прикладной
математики и информатики**

Кафедра прикладной математики
Практическое задание № 2+
по дисциплине «Структуры данных и алгоритмы»

СТРУКТУРЫ ДАННЫХ СТЕК И ОЧЕРЕДЬ

Бригада 2

ТАДЖИБАЕВ ЗАВКИДДИН

Группа ПМ-25

Вариант 4

Преподаватель

ТРАКИМУС ЮРИЙ ВИКТОРОВИЧ

Новосибирск, 2023

1 Задание

Используя стек решить следующую задачу:

В) В текстовом файле записано без ошибок логическое выражение (ЛВ) в следующей форме:

$$\langle \text{ЛВ} \rangle ::= t \mid f \mid (!\langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \& \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \mid \langle \text{ЛВ} \rangle)$$

Здесь знаки `!`, `&`, `|` обозначают соответственно отрицание, конъюнкцию и дизъюнкцию, а `t` – значение истина, `f` – значение ложь. Вычислить логическое значение этого выражения.

2 Анализ задания

Входные данные: В файл `in.txt` подаётся логическое выражение, соответствующее формуле:

$$\langle \text{ЛВ} \rangle ::= t \mid f \mid (!\langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \& \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \mid \langle \text{ЛВ} \rangle)$$

Выходные данные: Значение логического выражения; “Не удалось открыть файл `in.txt`.”; “Не удалось открыть файл `out.txt`.”; “Стек пуст”. В файл `out.txt` сохраняется значение выражение (ЛВ): “`t`” – true значение истина, “`f`” – false значение ложь.

Решение задачи: для решения задачи используется динамическая структура данных типа “Стек” для более эффективной работы с символами. В первую очередь программа записывает данные из файла в стек, после чего вызывается рекурсивная функция `process` которая поочерёдно обрабатывает символы из стека и вычисляет итоговый результат логического выражения.

3 Программа

```
#include <stdio.h>
#include <locale.h>
#include <windows.h>
#include <stdlib.h>

struct list
{
    char elem;
    list *next;
    list(char _elem = 0, list *_next = NULL):
        next(_next), elem(_elem) { };
};

struct stack
{
    list *l = NULL;
    inline bool empty() { return s->l == NULL; }
    void push(char c)
    {
        list *p = new list(c);
        p->next = s->l;
        s->l = p;
    }
    char pop()
    {
        if (s->empty())
            return NULL;
    }
};
```

```

        else
        {
            char c = NULL;

            list *p = s->l;
            c = p->elem;
            s->l = s->l->next;
            delete p;
            return c;
        }
    }
} *s = new stack;

bool input(stack *s)
{
    FILE *F = NULL;
    fopen_s(&F, "in.txt", "r");
    if (F)
    {
        for (char c = NULL; (c = fgetc(F)) != EOF; )
            s->push(c);
        return true;
    }
    else
        return false;
}

char process(stack *s, char val)
{
    for ( ; s->l; )
        switch (s->pop())
        {
            case '(':
                return val;
            case 't':
                {
                    val = 't';
                    break;
                }
            case 'f':
                {
                    val = 'f';
                    break;
                }
            case '&':
                {
                    val = val == 't' ? process(s, val) == 't' ? 't' : 'f' :
                        'f';
                    if (val == 'f') for ( ; s->pop() != '('; );
                    break;
                }
            case '|':
                {

```

```

        val = val == 'f' ? process(s, val) == 't' ? 't' : 'f':
        't';
        if (val == 't') for (; s->pop() != '('; );
        break;
    }
    case '!':
    {
        val == 'f' ? 't' : 'f';
        break;
    }
    default:
        break;
    }
    return val;
}

void output(char val)
{
    FILE *G = NULL;
    fopen_s(&G, "out.txt", "w");
    if (G)
    {
        fprintf_s(G, "%c", val);
        fclose(G);
    }
    else
        printf_s("Не удалось открыть файл out.txt.");
}

int main()
{
    setlocale(0, "");
    UINT cp = GetConsoleCP(), outcp = GetConsoleOutputCP();
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    char val = NULL;

    if (input(s))
    {
        if (s->empty())
            printf_s("Стек пуст.");
        else
        {
            val = process(s, val);
            output(val);
        }
    }
    else
        printf_s("Не удалось открыть файл in.txt.");

    SetConsoleCP(cp);
    SetConsoleOutputCP(outcp);
}

```

4 Набор тестов

| № | Входные данные | Назначение |
|----|--|--|
| 1 | | Файл отсутствует |
| 2 | | Файл пуст |
| 3 | ((t&f)&(t&t)) | Логическое выражение конъюнкции |
| 4 | ((t f) (t t) (t f)) | Логическое выражение дизъюнкции |
| 5 | (!(!(t))) | Логическое выражение отрицание |
| 6 | ((t&f) (t t) (((f t)&(t&f)))) | Логическое выражение дизъюнкции и конъюнкции |
| 7 | (!(!(t f) (t t) (((f t) (f f)))) (t f))) | Логическое выражение отрицание дизъюнкции |
| 8 | (!(!(t f) (((t&f)&(t&t))&(((f&t)&(f&f))))&(t&f)))&(((t&t)&(t&f))) | Логическое выражение отрицание конъюнкции |
| 9 | (!(!(t f) (((t&f)&(t t))&(((f&t) (f&f)))) (t&f)))&(((t t)&(t&f)) (t f))) | Логическое выражение отрицание дизъюнкции и конъюнкции |
| 10 | ((!(t f) (((t&f)&(t t))&(((f&t) (f&f)))) (t&f)))&(((t t)&(t&f)) (t f)))&(((t t)&(t&f)) (t&f))) | Логическое выражение |

5 Результаты работы программы

| № | Ввод/Вывод программы |
|----|---------------------------------|
| 1 | Не удалось открыть файл in.txt. |
| 2 | Стек пуст. |
| 3 | f |
| 4 | t |
| 5 | f |
| 6 | t |
| 7 | t |
| 8 | f |
| 9 | t |
| 10 | t |

