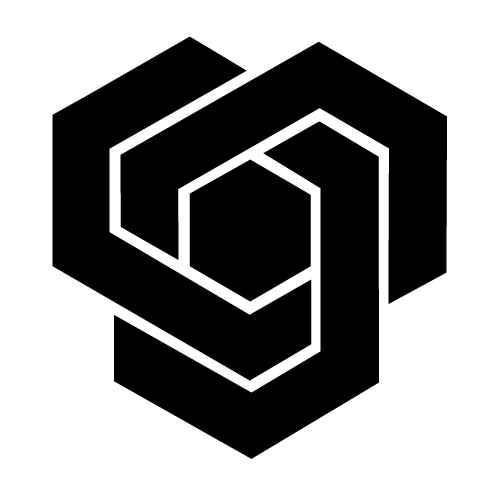
Технически университет – София



Курсова работа по ПИК 2

Изработил: Теодор Стилианов Добрев; 121217100; 38гр.

Преподавател: Явор Томов

Подпис:

Дата: 25.05.2018г.

*Задание No*  **25 дисциплина: ПИК II – курсова задача**

Теодор Стилианов Добрев

***Студент:***

**Да се разработи програма с меню тип пощенска кутия:**

1. Писмата на даден потребител се съхраняват в двоичен файл. Всяко писмо съдържа следните данни:

* Уникален идентификационен номер
* Получател
* Копие до
* Подател
* Заглавие
* Съдържание
* Дата на получаване - записана във формата ДД.ММ.ГГГГ HH:MM:SS.

2. Програмата трябва да предоставя на потребителя следната функционалност:

а) добавяне нов мейл към списъка;

б) показване на опростен списък с мейлите в пощенската кутия: номер, подател, заглавие, дата на получаване.  
в) показване на съдържанието на мейл по даден идентификационен номер;  
г) Изчиства всички спам писма. За спам писмо се приема писмо, в което съотношението на забранените думи към общия брой думи е повече от 1/5-та. Забранените думи са дефинирани в текстовия файл: „bad\_words.txt”;

3. Данните да се поддържат в динамична структура - едносвързан списък в оперативната памет на ПК.

***ИЗИСКВАНИЯ КЪМ ОФОРМЛЕНИЕТО***

Задачата да се оформи като задача, съдържаща:

* титулна страница с данни за студента, ръководителя на курсовата задача;
* текст на заданието;
* обобщен блоков алгоритъм на разработеното програмно осигуряване;
* описание на използуваните модули (функции) - прототип, входно изходни параметри и предназначение;
* общо описание за функциониране на програмата (вход/изход);
* листинг на source (изходния) код на програмата;
* резултати от изпълнението на програмата (контролен пример);
* проектът да се реализира в програмната среда като проект с разделна компилация.

**Дата на задаване: Преподавател:**

/………………………….. /

Програмата съдържа следните файлове:

* main.c – Съдържа функционалността на менюто по задачата
* mail.c – Функционалността на програмата се изпълнява тук
* mail.h – Свързва mail.c с main.c
* binary\_file\_mail\_list.bin – запазва свързания списък
* bad\_words.txt – Речник с всички думи, идентифицирани като „спам“

Описание на използваните модули в main.c :

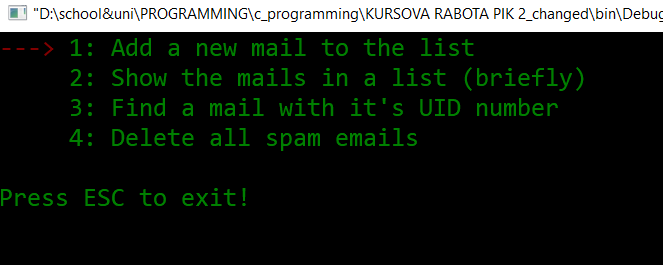
* int main() – Пуска менюто в началото на програмата
* void menu() – Занимава се с изписването на менюто и избора на потребителя
* void arrow\_or\_not (int current\_position) – Решава дали да изпише стрелката пред съответната опция в менюто за дадена опция(int current\_position)
* void arrow\_mover() – при даден калвиш, който променя позицията на стрелката(1-4 и UP, DOWN), се спомена глобалната променлива за позиция: int arrow\_position
* void function\_caller() – При ENTER на опция от менюто се извиква тази функция. Тя извиква избраната от потребителя функция за съответната опция и се грижи за това потрябитяля да прочете резултатите от избора си и СЛЕД това да се върне към менюто

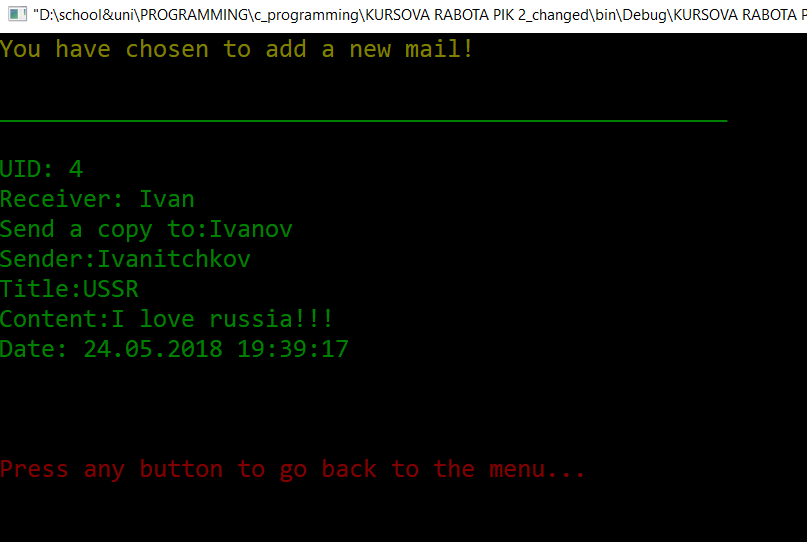
Описание на използваните модули в mail.c :

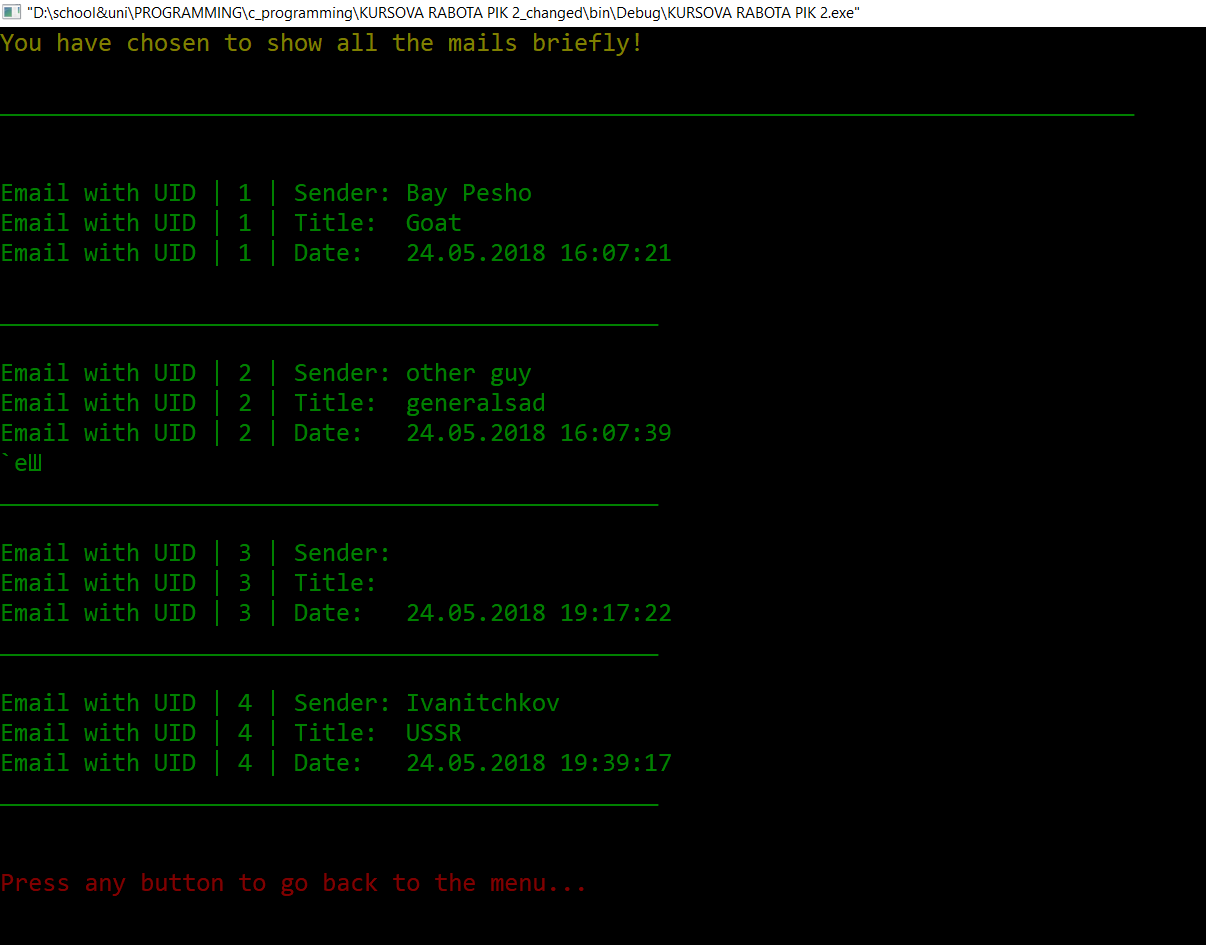
* char \* small\_letters(char str[264]) – Взима символен масив и го lowercase-ва; връща резултата
* void free\_linked\_list (Node \*head) – Освобождава паметта, алокирана за свързания списък
* void deleteNode(struct Node \*\*head, struct Node \*current) – Изтрива точно определен елемент от свързания списък; current – е за изтриване, а head е началото на свързания списък
* FILE \*open\_file(char\* mode) – Отваря файла в определен mode, ако не успее се опитва да го създаде и после да го отвори в дадения mode; Връща указателя към файла.
* void add\_element (Node \*\*add\_mail\_object, int i) – Въвеждане от потребителя на нов елемент (add\_mail\_object) към свързаният списък с уникален идентификатор (UID) като целочислено число “i”
* int is\_empty(FILE \*fb, int boolean) – Проверява дали файла (fb) е празен, връща 1 за празен и 0 за пълен; Boolean решава дали да се изпише съобщение дали файла е празен(1 – да се изпише)
* void add\_mail() – добавя новия елемент от add\_element опашъчно към свързани списък, ако има такъв или направо го записва ако е първият елемент на свързания списък (тоест бинарния файл е празен)
* void reveal\_mail() – Показва опростено всички имейли от бинарния файл
* void find\_by\_UID(int id) – Показва подробно всичко за имейла с UID, което съвпада с id-то, което е зададено от потребителя още в main.c
* int delete\_spam() – Премахва всички писма, в които има повече от 1/5 съдържание на спам думи, дефинирани и текстовия файл bad\_words.txt

КОНТРОЛЕН ПРИМЕР:

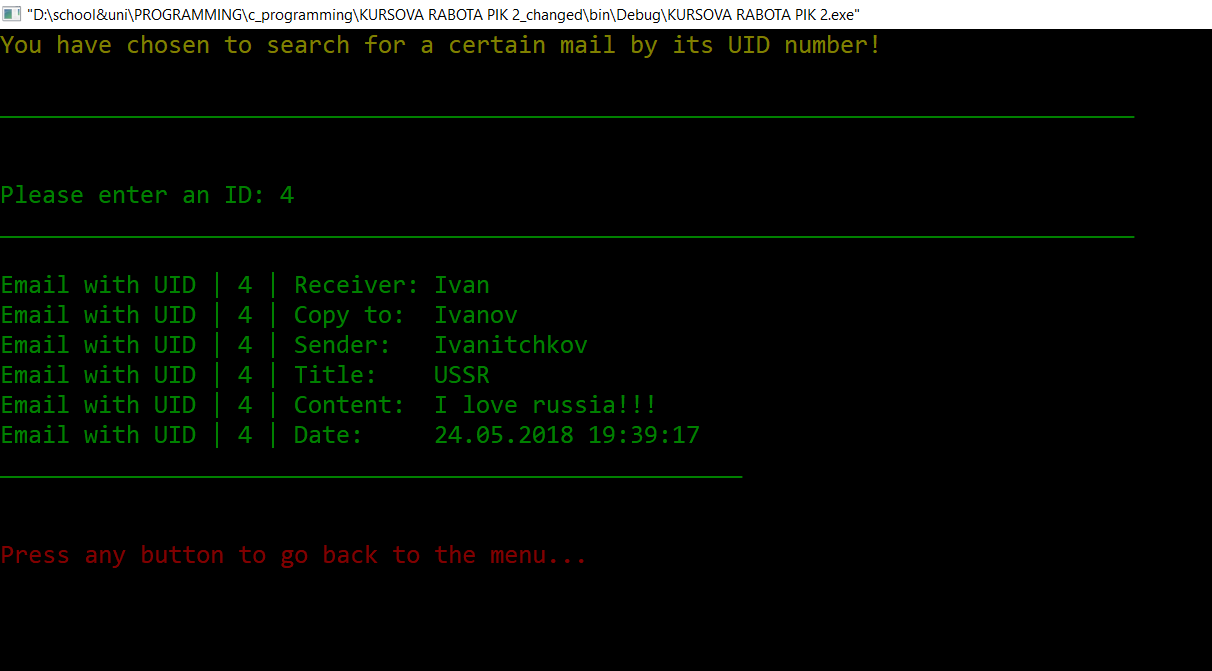
* меню:



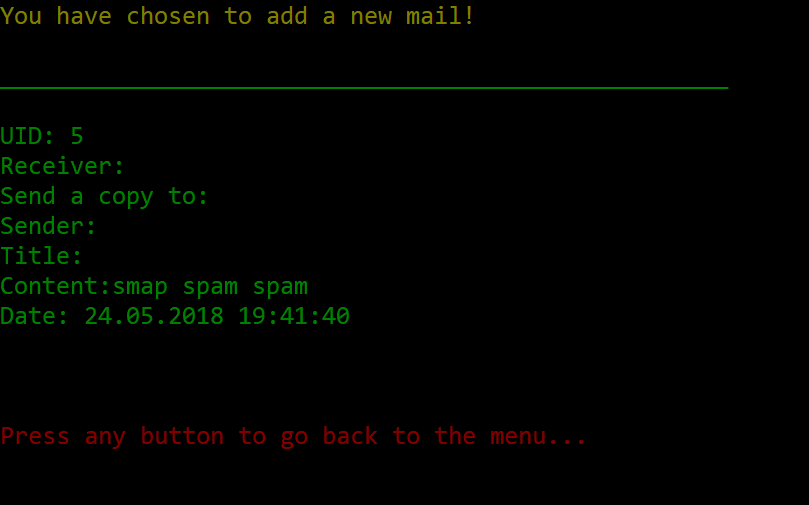
* опция 1:
* опция 2:



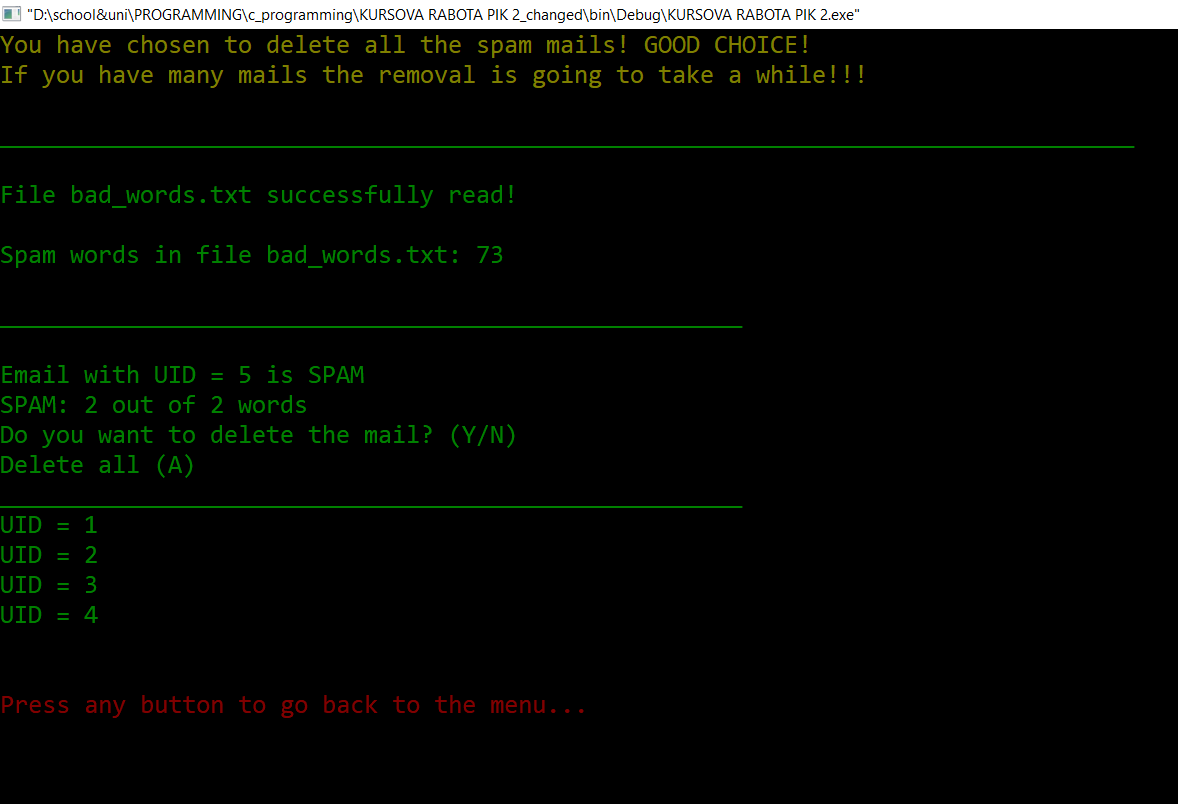
* опция 3:



* опция 1 със спам:



* опция 4 – за да премахне спам имейла:



* опция 2 – за проверка дали е изтрит спам имейла:

