**Отчет по лабораторной работе №10-11** по курсу \_\_Фундаментальная информатика\_\_\_\_\_\_\_\_\_\_\_\_

Студент группы М80-101Б-21 Постнов Александр Вячеславович, № по списку 17

Контакты www, e-maiSl:

Работа выполнена: «» 202 1г.

Преподаватель: каф. 806 \_\_\_\_\_Титов В.К.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Отчет сдан « » \_\_\_\_\_\_\_\_\_2021\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Тема:** \_Простейшие приёмы лексического анализа. Обработка лит последовательности текстового файла. Отладчик gdb\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Цель работы:** составить и отладить программу на СИ. Изучить основные команды отладчика gdb\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Задание** (*вариант: 17*)**:** Выписать все числа, записанные в троичной системе счисления и кратные 3
2. **Оборудование(лабораторное)**:

ЭВМ -, процессор -, имя узла сети - с ОП - ГБ,

НМД - ГБ, терминал- адрес -, принтер -

Другие устройства -

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 5 4500U, с ОП 8 ГБ

Другие устройства -

1. **Программное обеспечение:**

Операционная система семейства -, наименование - версия - интерпретатор команд - версия

Система программирования - версия -

Редактор текстов - версия -

Утилиты операционной системы -

Прикладные системы и программы -

Местонахождение и имена файлов программ и данных -

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства GNU/Linux, наименование Pop!\_OS версия 21.04

интерпретатор команд GNOME Terminal версия 3.38.2

Система программирования \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Редактор текстов emacs версия 3.27.20

Утилиты операционной системы cat

Прикладные системы и программы -

Местонахождение и имена файлов программ и данных -

1. **Идея, метод, алгоритм**  решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

1) Будем считывать символы из файла до тех пор, пока не встретим первую цифру

2) Проверим символ перед этой цифрой, если там есть минус, то учтем это

3) Считываем цифры, проверяем, что цифры от 0 до 2 (условие троичной системы счисления)

4) Переводим из char в int с помощью вычитания константы «0», сборка числа через умножение на 10.

5) Число в троичной системе счисления кратно 3, если оно «круглое», значит оно оканчивается на «0»

6) Если все условия соблюдены, распечатываем число на экран.

**7. Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

#include <stdio.h>

int only3(char c){ //удовлетворение троичной системы счисления

return (c >= '0' && c < '3');

}

int main(void){

int i = 0;

char c = '+'; //на случай, если число будет находиться в начале файла и без минуса

while (i != EOF){

int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

i = getchar();

if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

flag\_minus = 1;

}

c = i;

//printf("%c \n", c);

}

flag += !only3(c);

int num = c - '0';

while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

i = getchar();

c = i;

if (i < '0' || i > '9'){

break;

}

flag += !only3(c);

num = num \* 10 + c - '0'; //сборка числа

}

if (flag\_minus){

num = -num;

}

//printf("%d \n", num);

if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

printf("%d \n", num);

}

}

return 0;

}

Тесты:

Рассмотрим крайний случай, когда числа находятся в начале и в конце(оба подходят):

1010dfidjfsannfnfjnjfndfndfdfffjndnfjdfndfjfn

fdkfnnfkfnknkfkfdkffnfdkfnkdndkfdnfdnfdnkfnkfnkdfnkd1020

Ответ: 1010, 1020

Рассмотрим случай, когда в тексте есть числa, но они не подходят:

hello world, 456700, 22121, 221211re, tropa.

Ответ:

Рассмотрим случай, когда в тексте есть минусы и ответ – это число с минусом:

-3222332, geroy, rerrere,wqqwwqowwqio

wq-owwiqowio-wqoowoq-wioqwo-1002010ttt

Ответ: -1002010

С помощью отладчика gdb проверю случай:

h453t10

Ответ: 10

*Допущен к выполнению работы.*  **Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**8. Распечатка протокола**  (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

[alex@alex 10-11gdb(?)]$ cat head11.txt

-----------------------------------------------------

| Лабораторная работа №11 |

| Простейшие приёмы лексического анализа |

| Выполнил: студент группы М8О-101Б-21 |

| Постнов Александр Вячеславович |

-----------------------------------------------------

[alex@alex 10-11gdb(?)]$ cat main.c

#include <stdio.h>

int only3(char c){ //удовлетворение троичной системы счисления

return (c >= '0' && c < '3');

}

int main(void){

int i = 0;

char c = '+'; //на случай, если число будет находиться в начале файла и без минуса

while (i != EOF){

int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

i = getchar();

if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

flag\_minus = 1;

}

c = i;

//printf("%c \n", c);

}

flag += !only3(c);

int num = c - '0';

while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

i = getchar();

c = i;

if (i < '0' || i > '9'){

break;

}

flag += !only3(c);

num = num \* 10 + c - '0'; //сборка числа

}

if (flag\_minus){

num = -num;

}

//printf("%d \n", num);

if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

printf("%d \n", num);

}

}

return 0;

}

[alex@alex 10-11gdb(?)]$ g++ main.c

[alex@alex 10-11gdb(?)]$ cat test1.txt

1010dfidjfsannfnfjnjfndfndfdfffjndnfjdfndfjfn

fdkfnnfkfnknkfkfdkffnfdkfnkdndkfdnfdnfdnkfnkfnkdfnkd1020

[alex@alex 10-11gdb(?)]$ ./a.out <test1.txt

1010

1020

[alex@alex 10-11gdb(?)]$ cat test2.txt

hello world, 456700, 22121, 221211re, tropa.

[alex@alex 10-11gdb(?)]$ ./a.out <test2.txt

[alex@alex 10-11gdb(?)]$ cat test3.txt

-3222332, geroy, rerrere,wqqwwqowwqio

wq-owwiqowio-wqoowoq-wioqwo-1002010ttt

[alex@alex 10-11gdb(?)]$ ./a.out <test3.txt

-1002010

[alex@alex 10-11gdb(?)]$ cat head10.txt

-----------------------------------------------------

| Лабораторная работа №10 |

| Отладчик gdb |

| Выполнил: студент группы М8О-101Б-21 |

| Постнов Александр Вячеславович |

-----------------------------------------------------

[alex@alex 10-11gdb(?)]$ cat main.c

#include <stdio.h>

int only3(char c){ //удовлетворение троичной системы счисления

return (c >= '0' && c < '3');

}

int main(void){

int i = 0;

char c = '+'; //на случай, если число будет находиться в начале файла и без минуса

while (i != EOF){

int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

i = getchar();

if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

flag\_minus = 1;

}

c = i;

//printf("%c \n", c);

}

flag += !only3(c);

int num = c - '0';

while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

i = getchar();

c = i;

if (i < '0' || i > '9'){

break;

}

flag += !only3(c);

num = num \* 10 + c - '0'; //сборка числа

}

if (flag\_minus){

num = -num;

}

//printf("%d \n", num);

if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

printf("%d \n", num);

}

}

return 0;

}

[alex@alex 10-11gdb(?)]$ g++ -g main.c -o test

[alex@alex 10-11gdb(?)]$ cat gdb.txt

h453t10

[alex@alex 10-11gdb(?)]$ gdb test

GNU gdb (GDB) 11.1

Copyright (C) 2021 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

Type "show copying" and "show warranty" for details.

This GDB was configured as "x86\_64-pc-linux-gnu".

Type "show configuration" for configuration details.

For bug reporting instructions, please see:

<https://www.gnu.org/software/gdb/bugs/>.

Find the GDB manual and other documentation resources online at:

<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".

Type "apropos word" to search for commands related to "word"...

Reading symbols from test...

(gdb) list

1 #include <stdio.h>

2

3 int only3(char c){ //удовлетворение троичной системы счисления

4 return (c >= '0' && c < '3');

5 }

6

7 int main(void){

8 int i = 0;

9 char c = '+'; //на случай, если число будет находиться в начале файла и без минуса

10 while (i != EOF){

(gdb) b 7

Breakpoint 1 at 0x1177: file main.c, line 8.

(gdb) r <gdb.txt

Starting program: /home/alex/Рабочий стол/mai\_labs/10-11gdb(?)/test <gdb.txt

Breakpoint 1, main () at main.c:8

8 int i = 0;

(gdb) s

9 char c = '+'; //на случай, если число будет находиться в начале файла и без минуса

(gdb) print c

$1 = 0 '\000'

(gdb) s

10 while (i != EOF){

(gdb) s

11 int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

(gdb) s

12 int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

(gdb) print flag

$2 = 0

(gdb) ptype flag

type = int

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

14 i = getchar();

(gdb) print i

$3 = 0

(gdb) s

15 if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

(gdb) s

18 c = i;

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

14 i = getchar();

(gdb) s

15 if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

(gdb) s

18 c = i;

(gdb) print c

$4 = 104 'h'

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

21 flag += !only3(c);

(gdb) print flag

$5 = 0

(gdb) print c

$6 = 52 '4'

(gdb) s

only3 (c=52 '4') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) print flag

No symbol "flag" in current context.

(gdb) s

5 }

(gdb) s

main () at main.c:21

21 flag += !only3(c);

(gdb) print flag

$7 = 0

(gdb) s

22 int num = c - '0';

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb) s

24 i = getchar();

(gdb) s

25 c = i;

(gdb) print c

$8 = 52 '4'

(gdb) s

26 if (i < '0' || i > '9'){

(gdb) s

29 flag += !only3(c);

(gdb) s

only3 (c=53 '5') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) s

5 }

(gdb) s

main () at main.c:29

29 flag += !only3(c);

(gdb) print flag

$9 = 1

(gdb) s

30 num = num \* 10 + c - '0'; //сборка числа

(gdb) print num

$10 = 4

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb) s

24 i = getchar();

(gdb) s

25 c = i;

(gdb) print c

$11 = 53 '5'

(gdb) s

26 if (i < '0' || i > '9'){

(gdb) s

29 flag += !only3(c);

(gdb) s

only3 (c=51 '3') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) s

5 }

(gdb) s

main () at main.c:29

29 flag += !only3(c);

(gdb) s

30 num = num \* 10 + c - '0'; //сборка числа

(gdb) print num

$12 = 45

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb) s

24 i = getchar();

(gdb) s

25 c = i;

(gdb) s

26 if (i < '0' || i > '9'){

(gdb) s

32 if (flag\_minus){

(gdb) s

36 if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

(gdb) s

10 while (i != EOF){

(gdb) s

11 int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

(gdb) s

12 int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

14 i = getchar();

(gdb) s

15 if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

(gdb) s

18 c = i;

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

21 flag += !only3(c);

(gdb) s

only3 (c=49 '1') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) s

5 }

(gdb) s

main () at main.c:21

21 flag += !only3(c);

(gdb) s

22 int num = c - '0';

(gdb) print num

$13 = 453

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb)

24 i = getchar();

(gdb) s

25 c = i;

(gdb) print c

$14 = 49 '1'

(gdb) s

26 if (i < '0' || i > '9'){

(gdb) s

29 flag += !only3(c);

(gdb) s

only3 (c=48 '0') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) s

5 }

(gdb) s

main () at main.c:29

29 flag += !only3(c);

(gdb) s

30 num = num \* 10 + c - '0'; //сборка числа

(gdb) print num

$15 = 1

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb) s

24 i = getchar();

(gdb) s

25 c = i;

(gdb) s

26 if (i < '0' || i > '9'){

(gdb) s

32 if (flag\_minus){

(gdb) s

36 if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

(gdb) s

37 printf("%d \n", num);

(gdb) print num

$16 = 10

(gdb) s

10

10 while (i != EOF){

(gdb) s

11 int flag\_minus = 0; //присваивается 1, если перед числом есть знак -

(gdb) s

12 int flag = 0; //прибавляется значения к флагу, если в числе есть цифры >= 3

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

14 i = getchar();

(gdb) s

15 if (i >= '0' && i <= '9' && c == '-'){ //следующий символ -- цифра, значит проверяем текущий символ на наличие минуса

(gdb) s

18 c = i;

(gdb) s

13 while (i != EOF && (i < '0' || i > '9')){ //до первой цифры

(gdb) s

21 flag += !only3(c);

(gdb) s

only3 (c=-1 '\377') at main.c:4

4 return (c >= '0' && c < '3');

(gdb) s

5 }

(gdb) s

main () at main.c:21

21 flag += !only3(c);

(gdb) s

22 int num = c - '0';

(gdb) s

23 while (i != EOF && i >= '0' && i <= '9'){ //до конца числа

(gdb) s

32 if (flag\_minus){

(gdb) s

36 if (flag == 0 && num % 10 == 0){ //круглые числа в троичной системе счисления кратные 3

(gdb) s

10 while (i != EOF){

(gdb) s

40 return 0;

(gdb) s

41 }

(gdb) s

0x00007ffff7a8ab25 in \_\_libc\_start\_main () from /usr/lib/libc.so.6

(gdb) s

Single stepping until exit from function \_\_libc\_start\_main,

which has no line number information.

[Inferior 1 (process 6822) exited normally]

(gdb) quit

[alex@alex 10-11gdb(?)]$

**9. Дневник отладки** должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
| 1 | дом | 14.11 | 13:51 | Программа не скомпилировалась из-за синтаксической ошибки | Добавил “;” в строках 24, 25. |  |

**10. Замечания автора :**

11.**Выводы**

\_\_В ходе лабораторной работе я составил и отладил программу на СИ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_Постнов\_\_\_\_\_\_\_\_\_\_\_\_\_\_