Министерство образования Республики Беларусь

Белорусский Национальный Технический Университет

Факультет информационных технологий и робототехники

Кафедра «Программное обеспечение информационных систем и технологий»

**Отчёт**

***по Лабораторной работе №25***

по дисциплине ***«Языки программирования»***

Исполнитель: Колосов Артём Александрович

Группа: 10701219

Преподаватель: Лапанович Ирина Олеговна

2020

Цель работы: Закрепить навыки работы с текстовыми данными и файлами в языке Python.

Основное задание:

1) Дан текстовый файл in.txt, в котором содержится достаточно большой отрывок из Вашего любимого романа (можно как на английском языке, так и на русском). В качестве контрольного примера ниже дана глава «The Potions Master» из книги о Гарри Поттере «Harry Potter and the Sorcerer's Stone» (автор J.K. Rowling). Необходимо написать программу, которая бы разбирала целевой текст на слова, подсчитывала вхождение каждого слова, сортировала результат по убыванию/возрастанию и записывала результат в файл out.txt.

2) Дан файл important.in, в котором зашифрован текст, означающий БОЛЬШУЮ истину для начинающих разработчиков: vujgvmCfb tj ufscfu ouib z/vhm jdjuFyqm jt fscfuu uibo jdju/jnqm fTjnqm tj scfuuf ibou fy/dpnqm yDpnqmf jt cfuufs boui dbufe/dpnqmj uGmb tj fuufsc ouib oftufe/ bstfTq jt uufscf uibo otf/ef uzSfbebcjmj vout/dp djbmTqf dbtft (ubsfo djbmtqf hifopv up csfbl ifu t/svmf ipvhiBmu zqsbdujdbmju fbutc uz/qvsj Fsspst tipvme wfsof qbtt foumz/tjm omfttV mjdjumzfyq odfe/tjmf Jo fui dfgb pg hvjuz-bncj gvtfsf fui ubujpoufnq up ftt/hv Uifsf vmetip fc pof.. boe sbcmzqsfgf zpom pof pvt..pcwj xbz pu pe ju/ Bmuipvhi uibu bzx bzn puo cf wjpvtpc bu jstug ttvomf sfzpv( i/Evud xOp tj scfuuf ibou /ofwfs uipvhiBm fsofw jt fopgu cfuufs boui iu++sjh x/op gJ ifu nfoubujpojnqmf tj eibs pu mbjo-fyq tju( b bec /jefb Jg fui foubujpojnqmfn jt fbtz up bjo-fyqm ju znb cf b hppe jefb/ bnftqbdftO bsf pof ipoljoh sfbuh efbj .. fu(tm pe psfn gp tf"uip Необходимо разработать алгоритм и реализовать программу, которая бы смогла восстановить исходный текст и записать его в файл important.out. Python File and Text Handling

3) Дан файл code.in исходного кода на языке программирования Python. Необходимо написать программу, которая создавала бы копию данного исходного кода в виде файла code.out и удаляла в нём комментарии.

Индивидуальное задание:

Необходимо выполнить задания из раздела «Работа со строками», которые даны на лабораторной работе по теме «Строки и файлы» по параллельной специализированной дисциплине «Основы алгоритмизации и программирования», но с одним дополнением – все строковые данные берутся из файла.

Требования к выполнению:

1) Необходимо спроектировать блок-схемы алгоритмов решений соответствую-

щих заданий согласно своему варианту (варианты назначаются преподавате-

лем или выбираются самостоятельно студентом, но ни один вариант не дол-

жен повторяться у студентов из одной группы) и на базе данных алгоритмов

разработать интерактивные консольные приложения с использованием архи-

тектурного шаблона проектирования ***MVC***.

2) Для каждого алгоритма решения задания оценить (вычислить) алгоритмиче-

скую сложность по необходимому времени и затрачиваемой памяти исполь-

зуя определение ***Big O***.

3) Для масштабируемости разрабатываемого программного решения размер

последовательности (списка) и его элементы должны задаваться пользовате-

лем во время выполнения программы или с помощью генератора псевдослу-

чайных чисел.

4) Для автоматизации заполнения различными значениями искомого контей-

нера рекомендуется использовать соответствующие функции генерирования

псевдослучайных чисел модуля ***random***.

5) Рекомендуется избегать использования глобальных переменных при написа-

нии основной логики приложения.

6) Если логически не подразумевается или в заданиях иного не указано, то вход-

ными и выходными данными являются вещественные числа (числа с плаваю-

щей запятой).

7) Все программы должны быть разбиты на отдельные функции. Среди данных

функций рекомендуется добавлять стартовую функцию ***main***, с которой лучше

производить запуск программы.

8) При выполнении заданий необходимо по максимуму пытаться разрабатывать

универсальный, масштабируемый, легко поддерживаемый и читаемый код.

9) Также рекомендуется придерживаться ***Single Responsibility Principle****,* ***SRP***

(принципа единственной ответственности) – постарайтесь вынести основную

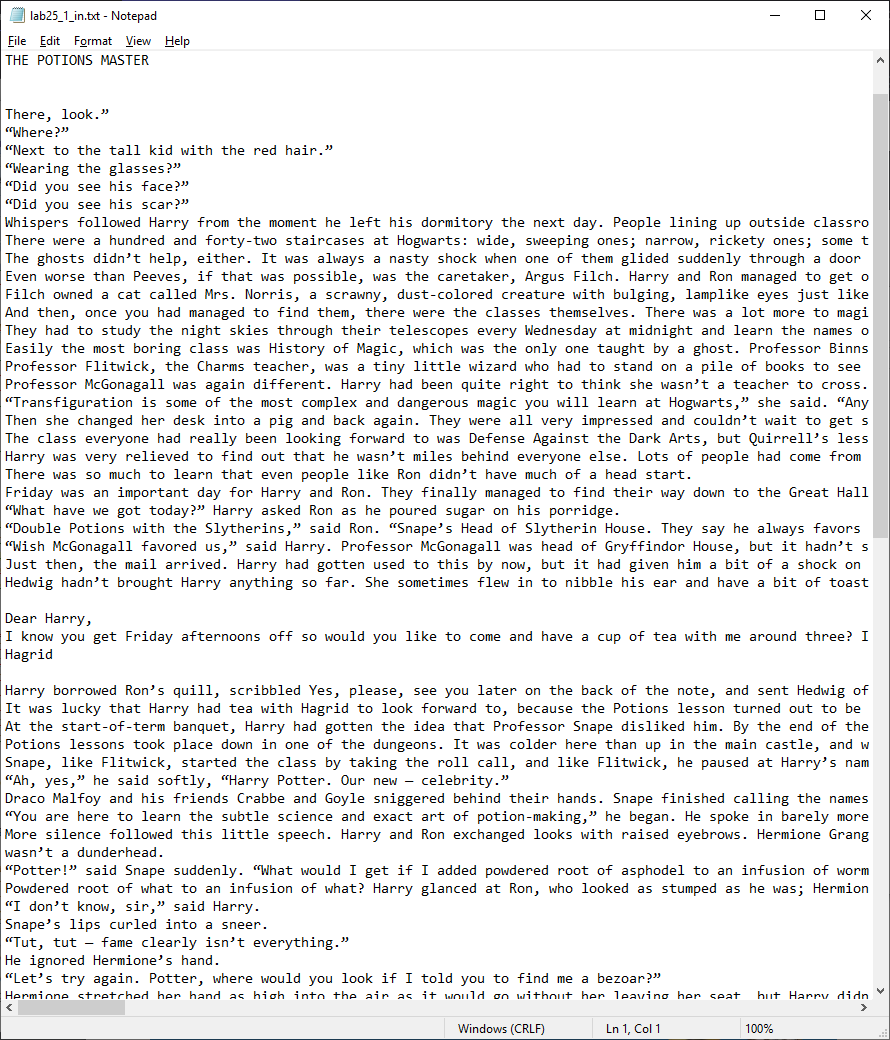
бизнес-логику задания в отдельную функцию (т.е. архитектура приложения

должна минимум состоять из нескольких функций).

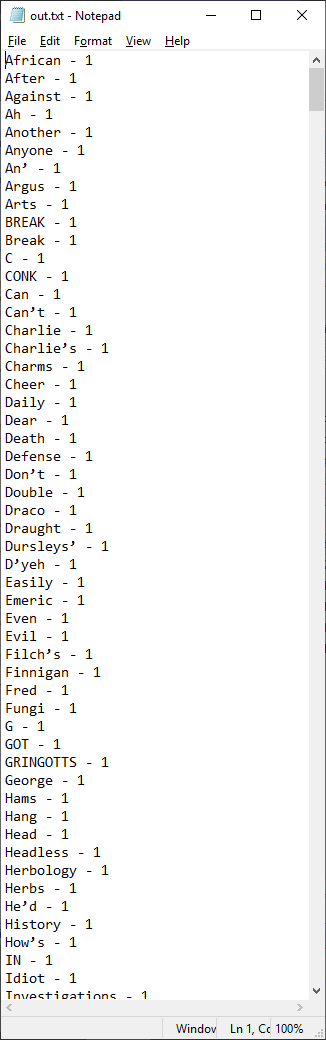
10) В соответствующих компонентах (функциях) бизнес-логики необходимо

предусмотреть «защиту от дурака».

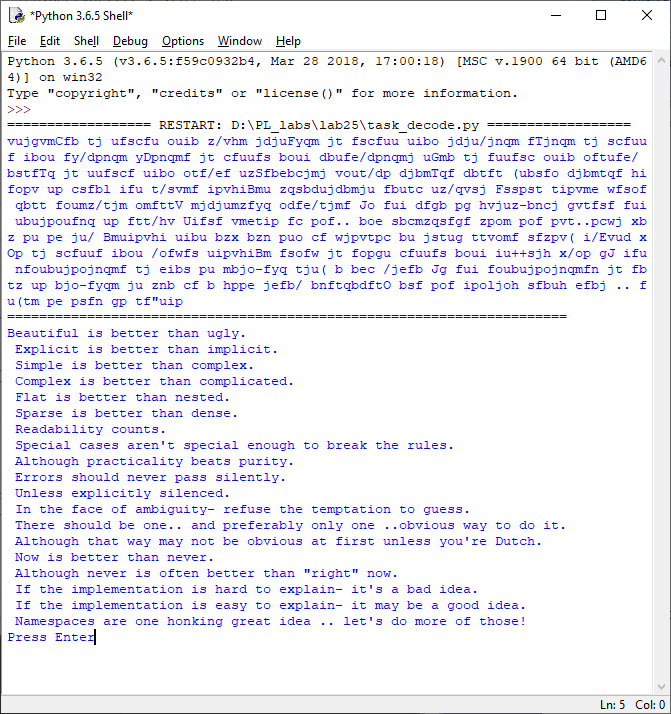
Результаты выполнения основного задания:



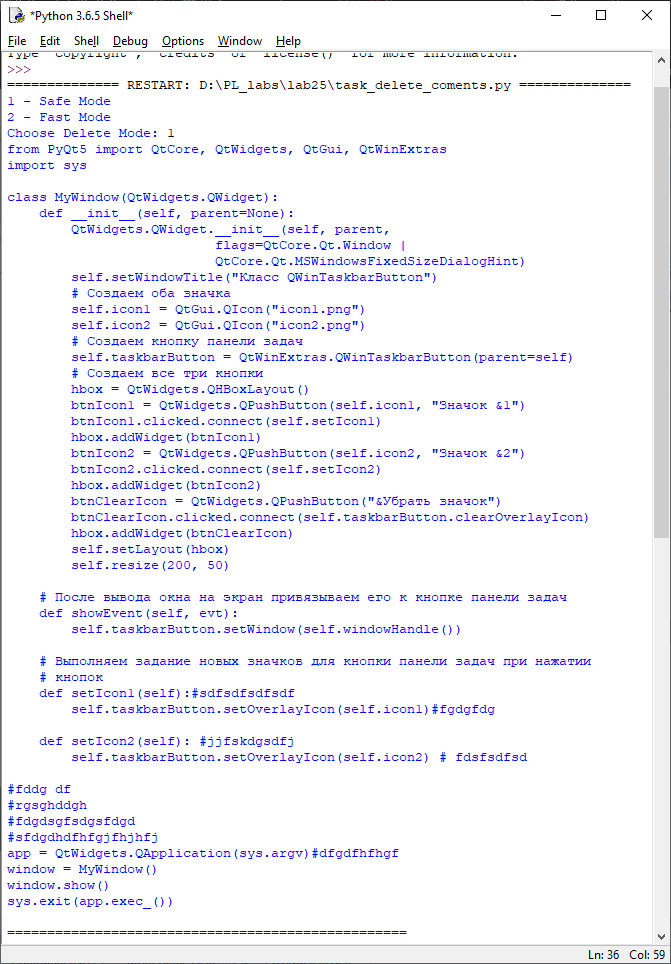
**Рис 25.1 Пример содержания текстового файла из 1-го задания**



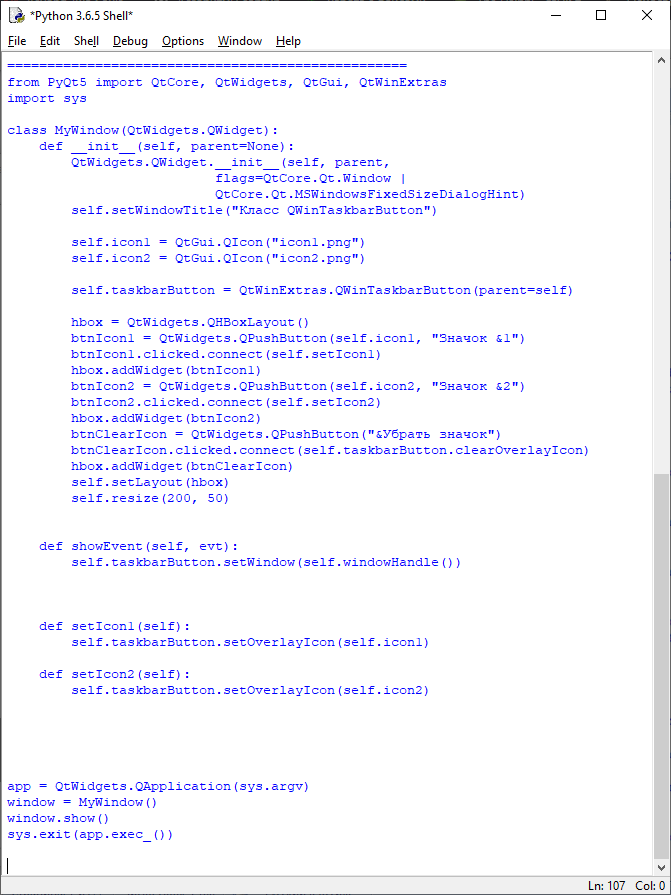
**Рис 25.2 Содержание текстового файла с результатами работы из 1-го задания**



**Рис 25.3 Результат работы 2-го основного задания**

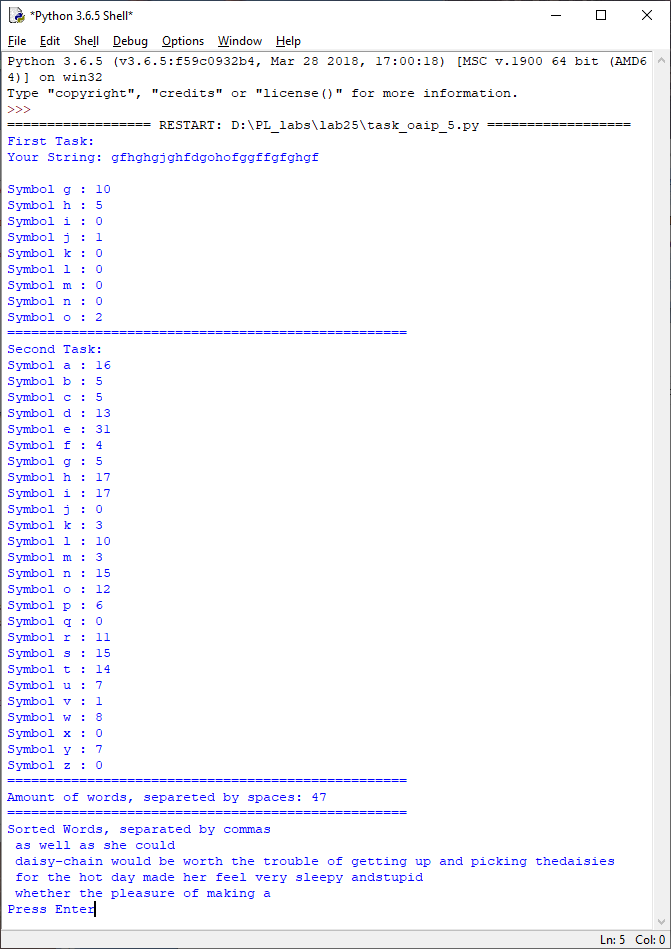


**Рис 25.4 Результат работы 3-го основного задания (До удаления комментариев)**

****

**Рис 25.5 Результат работы 3-го основного задания (После удаления комментариев)**

Результаты выполнения индивидуального задания:

****

**Рис 25.6 Результат работы индивидуального задания**

Что я освоил в процессе выполнения лабораторной работы (выводы):

Мной были закреплены знания по работе с файлами в языке программирования Python. Также были созданы приложения по работе с файлами, который имеют довольно большую практическую ценность.

*ПРИЛОЖЕНИЕ A*

Листинг исходных кодов программ

**Файл task\_decode.py**

# Программа расшифровывает закодированное сообщение из файла

#

# Version: 1.0

# Group: 10701219

# Author: Колосов Артём Александрович

# Date: 16.4.2020

def decrypt\_сaesar(message, shift=1):

if not isinstance(message, str):

return None

alpha\_lower = [chr(i) for i in range(ord('a'), ord('z') + 1)]

alpha\_upper = [chr(i) for i in range(ord('A'), ord('Z') + 1)]

result = ""

for symbol in message:

try:

if symbol in alpha\_lower:

ind = alpha\_lower.index(symbol)

step = ind - shift

result += alpha\_lower[step]

elif symbol in alpha\_upper:

ind = alpha\_upper.index(symbol)

step = ind - shift

result += alpha\_upper[step]

else:

result += symbol

except IndexError as msg:

print(msg)

break

return result

def find\_slice\_len(text):

if not isinstance(text, str):

return None

min\_index = 0

minimum = ord(text[0])

for i in range(len(text)):

if minimum > ord(text[i]):

min\_index = i

minimum = ord(text[i])

return len(text[min\_index:])

def decode\_additional(decrypt, slice\_len):

if not (isinstance(decrypt, list) and isinstance(slice\_len, int)):

return None

encrypted\_dot\_symbol = '/'

for i in range(len(decrypt)):

if len(decrypt[i]) > slice\_len:

res = len(decrypt[i]) - slice\_len

decrypt[i] = decrypt[i][res:len(decrypt[i])] + decrypt[i][0:res]

else:

res = -slice\_len % len(decrypt[i])

decrypt[i] = decrypt[i][res:] + decrypt[i][0:res]

if encrypted\_dot\_symbol in decrypt[i]:

slice\_len += 1

def main():

path = r"important.txt"

try:

file = open(path, "r")

except FileNotFoundError:

print("Error: File ", path, " NotFound")

text = file.read()

decrypt = decrypt\_сaesar(text, 1).split()

print(" ".join(decrypt))

slice\_len = find\_slice\_len(decrypt[0])

decode\_additional(decrypt, slice\_len)

decrypt = ' '.join(decrypt)

replace\_dict = {'(': '\'',

'/': '.\n',

'"': '!',

'+': '"'}

print("="\*70)

for key in replace\_dict.keys():

decrypt = decrypt.replace(key, replace\_dict[key])

print(decrypt)

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Файл task\_delete\_coments.py**

# Программа удаляет комментарии из кода Python

#

# Version: 1.0

# Group: 10701219

# Author: Колосов Артём Александрович

# Date: 16.4.2020

def delete\_comments\_safe\_way(code\_lines):

if not isinstance(code\_lines, list):

return None

for i in range(len(code\_lines)):

if isinstance(code\_lines[i], str):

flag = False

prom = list(code\_lines[i])

for j in range(len(prom)):

if prom[j] == '#':

flag = True

if flag and prom[j] != '\n':

prom[j] = ''

code\_lines[i] = "".join(prom)

def delete\_comments\_faster\_way(code\_lines):

if not isinstance(code\_lines, list):

return None

for i in range(len(code\_lines)):

if isinstance(code\_lines[i], str):

flag = False

prom = list(code\_lines[i])

start\_index = 0

for j in range(len(code\_lines[i])):

if prom[j] == '#':

flag = True

start\_index = j

break

if flag:

code\_lines[i] = code\_lines[i][0:start\_index] + '\n'

def main():

path\_in = r"code.in.txt"

path\_out = r"code.out.txt"

try:

file = open(path\_in, "r")

except FileNotFoundError:

print("Error: FileNotFound!")

exit(1)

print("1 - Safe Mode\n"

"2 - Fast Mode")

command = input("Choose Delete Mode: ")

all\_lines = file.readlines()

file.close()

print(''.join(all\_lines))

print("="\*50)

if command == "1":

delete\_comments\_safe\_way(all\_lines)

print(''.join(all\_lines))

elif command == "2":

delete\_comments\_faster\_way(all\_lines)

print(''.join(all\_lines))

new\_file = open(path\_out, "w")

new\_file.writelines(all\_lines)

new\_file.close()

input()

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Файл task\_main.py**

# Программа Подсчитывает кол-во повторений

# Каждого слова в тексте, и выводит результаты

# в новый файл

#

# Version: 1.0

# Group: 10701219

# Author: Колосов Артём Александрович

# Date: 16.4.2020

def output\_to\_user(msg, s=' ', e='\n'):

print(msg, sep=s, end=e)

def isalpha\_mod(string):

result = False

if isinstance(string, str):

result = str.isalpha(string) or string == "’" or string == "'"

return result

def get\_words(text):

if not isinstance(text, str):

return None

all\_words = {}

time\_text = []

flag = False

for symbol in text:

if isalpha\_mod(symbol):

time\_text.append(symbol)

flag = True

elif flag:

string = "".join(time\_text)

if all\_words.get(string):

all\_words[string] += 1

else:

all\_words[string] = 1

time\_text.clear()

flag = False

return all\_words

def main():

try:

path = r"lab25\_1\_in.txt"

file = open(path, "r")

except FileNotFoundError:

output\_to\_user("Error: FileNotFound!")

exit(1)

text = file.read()

text\_words = get\_words(text)

new\_file = open("out.txt", "w")

if isinstance(text\_words, dict):

repet = list(zip(text\_words.values(),text\_words.keys()))

repet.sort()

for i in range(0, len(repet)):

new\_file.write("{0} - {1}\n".format(repet[i][1], repet[i][0]))

new\_file.close()

input("Task Ended Sucessfully!Press Enter")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Файл task\_oaip\_5.py**

# Программа подсчитывает кол-во символов в диапазоне от 'g' до 'o'

# Подсчитывает кол-во каждого символа в тексте

# Находит предложения, разделённые с двух сторон запятыми и

# Сортирует их по алфавиту

# Считает кол-во слов, разделённых с двух сторон пробелами

# Version: 1.0

# Group: 10701219

# Author: Колосов Артём Александрович

# Date: 16.4.2020

def output\_to\_user(msg, s=' ', e='\n'):

print(msg, sep=s, end=e)

def count\_symbol\_codes(string, symb\_one='g', symb\_two='o'):

if not (isinstance(string, str) and isinstance(symb\_one, str) or isinstance(symb\_two, str)):

return None

count = 0

first\_code = ord(symb\_one)

second\_code = ord(symb\_two)

for symbol in string:

if first\_code <= ord(symbol) <= second\_code:

count += 1

return count

def count\_symbol\_alternate(string, symbol\_one='g', symbol\_two='o', key=lambda s: s):

if not (isinstance(string, str) and isinstance(symbol\_one, str) and isinstance(symbol\_two, str)):

return None

first\_code = ord(key(symbol\_one))

second\_code = ord(key(symbol\_two))

for code in range(first\_code, second\_code + 1):

output\_to\_user("Symbol {0} : {1}".format(chr(code), string.count(key(chr(code)))))

def selection\_sort(text):

if not isinstance(text, list):

return None

index = 0

for i in range(len(text) - 1):

index = i

for j in range(i + 1, len(text)):

if strcmp(text[j], text[index]) < 0:

index = j

if index != i:

text[i], text[index] = text[index], text[i]

return text

def strcmp(str1, str2):

if not (isinstance(str1, str) and isinstance(str2, str)):

return None

smallest\_len = len(str1) if len(str1) <= len(str2) else len(str2)

return\_code = 0

str1 = str1.strip()

str2 = str2.strip()

for i in range(0, smallest\_len):

if str1[i] != str2[i]:

return\_code = 1 if ord(str1[i]) > ord(str2[i]) else -1

break

return return\_code

def count\_spaces(my\_text):

if not isinstance(my\_text, str):

return None

count = 0

for i in range(len(my\_text)):

if my\_text[i].isspace():

flag = False

for j in range(i + 1, len(my\_text)):

if my\_text[j].isalpha():

flag = True

elif my\_text[j].isspace():

if flag:

count += 1

i += j

break

elif my\_text[j] in (",", ".", "?", ":", ";", "/"):

i += j

break

return count

def main():

split\_symbol = '+'

try:

file = open("oaip\_file.txt", "r")

info = file.read().split(split\_symbol)

phrase = info[0]

text = info[1]

flag = True

except FileNotFoundError as msg:

output\_to\_user("Error:{0}".format(msg))

input()

exit(1)

except IndexError as msg:

output\_to\_user("Error:{0}".format(msg))

input()

exit(1)

output\_to\_user("First Task:")

output\_to\_user("Your String: {0}".format(phrase))

count\_symbol\_alternate(phrase)

output\_to\_user("="\*50)

output\_to\_user("Second Task:")

count\_symbol\_alternate(text, key=lambda s: s.lower(), symbol\_one='a', symbol\_two='z')

text = text.replace("\n", "")

words = text.split(",")

words = words[1:len(words) - 1]

output\_to\_user("="\*50)

output\_to\_user("Amount of words, separeted by spaces: {0}".format(count\_spaces(text)))

output\_to\_user("="\*50)

output\_to\_user("Sorted Words, separated by commas")

words = selection\_sort(words)

for word in words:

output\_to\_user(word)

input("Press Enter")

if \_\_name\_\_ == "\_\_main\_\_":

main()

*ПРИЛОЖЕНИЕ Б*

Содержание текстовых файлов

Файл lab25.1.txt

J.K. Rowling - Harry Potter and the Sorcerer's Stone

C H A P T E R E I G H T

THE POTIONS MASTER

There, look.”

“Where?”

“Next to the tall kid with the red hair.”

“Wearing the glasses?”

“Did you see his face?”

“Did you see his scar?”

Whispers followed Harry from the moment he left his dormitory the next day. People lining up outside classrooms stood on tiptoe to get a look at him, or doubled back to pass him in the corridors again, staring. Harry wished they wouldn’t, because he was trying to concentrate on finding his way to classes.

There were a hundred and forty-two staircases at Hogwarts: wide, sweeping ones; narrow, rickety ones; some that led somewhere different on a Friday; some with a vanishing step halfway up that you had to remember to jump. Then there were doors that wouldn’t open unless you asked politely, or tickled them in exactly the right place, and doors that weren’t really doors at all, but solid walls just pretending. It was also very hard to remember where anything was, because it all seemed to move around a lot. The people in the portraits kept going to visit each other, and Harry was sure the coats of armor could walk.

The ghosts didn’t help, either. It was always a nasty shock when one of them glided suddenly through a door you were trying to open. Nearly Headless Nick was always happy to point new Gryffindors in the right direction, but Peeves the Poltergeist was worth two locked doors and a trick staircase if you met him when you were late for class. He would drop wastepaper baskets on your head, pull rugs from under your feet, pelt you with bits of chalk, or sneak up behind you, invisible, grab your nose, and screech, “GOT YOUR CONK!”

Even worse than Peeves, if that was possible, was the caretaker, Argus Filch. Harry and Ron managed to get on the wrong side of him on their very first morning. Filch found them trying to force their way through a door that unluckily turned out to be the entrance to the out-of-bounds corridor on the third floor. He wouldn’t believe they were lost, was sure they were trying to break into it on purpose, and was threatening to lock them in the dungeons when they were rescued by Professor Quirrell, who was passing.

Filch owned a cat called Mrs. Norris, a scrawny, dust-colored creature with bulging, lamplike eyes just like Filch’s. She patrolled the corridors alone. Break a rule in front of her, put just one toe out of line, and she’d whisk off for Filch, who’d appear, wheezing, two seconds later. Filch knew the secret passageways of the school better than anyone (except perhaps the Weasley twins) and could pop up as suddenly as any of the ghosts. The students all hated him, and it was the dearest ambition of many to give Mrs. Norris a good kick.

And then, once you had managed to find them, there were the classes themselves. There was a lot more to magic, as Harry quickly found out, than waving your wand and saying a few funny words.

They had to study the night skies through their telescopes every Wednesday at midnight and learn the names of different stars and the movements of the planets. Three times a week they went out to the greenhouses behind the castle to study Herbology, with a dumpy little witch called Professor Sprout, where they learned how to take care of all the strange plants and fungi, and found out what they were used for.

Easily the most boring class was History of Magic, which was the only one taught by a ghost. Professor Binns had been very old indeed when he had fallen asleep in front of the staff room fire and got up next morning to teach, leaving his body behind him. Binns droned on and on while they scribbled down names and dates, and got Emeric the Evil and Uric the Oddball mixed up.

Professor Flitwick, the Charms teacher, was a tiny little wizard who had to stand on a pile of books to see over his desk. At the start of their first class he took the roll call, and when he reached Harry’s name he gave an excited squeak and toppled out of sight.

Professor McGonagall was again different. Harry had been quite right to think she wasn’t a teacher to cross. Strict and clever, she gave them a talking-to the moment they sat down in her first class.

“Transfiguration is some of the most complex and dangerous magic you will learn at Hogwarts,” she said. “Anyone messing around in my class will leave and not come back. You have been warned.”

Then she changed her desk into a pig and back again. They were all very impressed and couldn’t wait to get started, but soon realized they weren’t going to be changing the furniture into animals for a long time. After taking a lot of complicated notes, they were each given a match and started trying to turn it into a needle. By the end of the lesson, only Hermione Granger had made any difference to her match; Professor McGonagall showed the class how it had gone all silver and pointy and gave Hermione a rare smile.

The class everyone had really been looking forward to was Defense Against the Dark Arts, but Quirrell’s lessons turned out to be a bit of a joke. His classroom smelled strongly of garlic, which everyone said was to ward off a vampire he’d met in Romania and was afraid would be coming back to get him one of these days. His turban, he told them, had been given to him by an African prince as a thank-you for getting rid of a troublesome zombie, but they weren’t sure they believed this story. For one thing, when Seamus Finnigan asked eagerly to hear how Quirrell had fought off the zombie, Quirrell went pink and started talking about the weather; for another, they had noticed that a funny smell hung around the turban, and the Weasley twins insisted that it was stuffed full of garlic as well, so that Quirrell was protected wherever he went.

Harry was very relieved to find out that he wasn’t miles behind everyone else. Lots of people had come from Muggle families and, like him, hadn’t had any idea that they were witches and wizards.

There was so much to learn that even people like Ron didn’t have much of a head start.

Friday was an important day for Harry and Ron. They finally managed to find their way down to the Great Hall for breakfast without getting lost once.

“What have we got today?” Harry asked Ron as he poured sugar on his porridge.

“Double Potions with the Slytherins,” said Ron. “Snape’s Head of Slytherin House. They say he always favors them — we’ll be able to see if it’s true.”

“Wish McGonagall favored us,” said Harry. Professor McGonagall was head of Gryffindor House, but it hadn’t stopped her from giving them a huge pile of homework the day before.

Just then, the mail arrived. Harry had gotten used to this by now, but it had given him a bit of a shock on the first morning, when about a hundred owls had suddenly streamed into the Great Hall during breakfast, circling the tables until they saw their owners, and dropping letters and packages onto their laps.

Hedwig hadn’t brought Harry anything so far. She sometimes flew in to nibble his ear and have a bit of toast before going off to sleep in the owlery with the other school owls. This morning, however, she fluttered down between the marmalade and the sugar bowl and dropped a note onto Harry’s plate. Harry tore it open at once. It said, in a very untidy scrawl:

Dear Harry,

I know you get Friday afternoons off so would you like to come and have a cup of tea with me around three? I want to hear all about your first week. Send us an answer back with Hedwig.

Hagrid

Harry borrowed Ron’s quill, scribbled Yes, please, see you later on the back of the note, and sent Hedwig off again.

It was lucky that Harry had tea with Hagrid to look forward to, because the Potions lesson turned out to be the worst thing that had happened to him so far.

At the start-of-term banquet, Harry had gotten the idea that Professor Snape disliked him. By the end of the first Potions lesson, he knew he’d been wrong. Snape didn’t dislike Harry — he hated him.

Potions lessons took place down in one of the dungeons. It was colder here than up in the main castle, and would have been quite creepy enough without the pickled animals floating in glass jars all around the walls.

Snape, like Flitwick, started the class by taking the roll call, and like Flitwick, he paused at Harry’s name.

“Ah, yes,” he said softly, “Harry Potter. Our new — celebrity.”

Draco Malfoy and his friends Crabbe and Goyle sniggered behind their hands. Snape finished calling the names and looked up at the class. His eyes were black like Hagrid’s, but they had none of Hagrid’s warmth. They were cold and empty and made you think of dark tunnels.

“You are here to learn the subtle science and exact art of potion-making,” he began. He spoke in barely more than a whisper, but they caught every word — like Professor McGonagall, Snape had the gift of keeping a class silent without effort. “As there is little foolish wand-waving here, many of you will hardly believe this is magic. I don’t expect you will really understand the beauty of the softly simmering cauldron with its shimmering fumes, the delicate power of liquids that creep through human veins, bewitching the mind, ensnaring the senses. . . . I can teach you how to bottle fame, brew glory, even stopper death — if you aren’t as big a bunch of dunderheads as I usually have to teach.”

More silence followed this little speech. Harry and Ron exchanged looks with raised eyebrows. Hermione Granger was on the edge of her seat and looked desperate to start proving that she

wasn’t a dunderhead.

“Potter!” said Snape suddenly. “What would I get if I added powdered root of asphodel to an infusion of wormwood?”

Powdered root of what to an infusion of what? Harry glanced at Ron, who looked as stumped as he was; Hermione’s hand had shot into the air.

“I don’t know, sir,” said Harry.

Snape’s lips curled into a sneer.

“Tut, tut — fame clearly isn’t everything.”

He ignored Hermione’s hand.

“Let’s try again. Potter, where would you look if I told you to find me a bezoar?”

Hermione stretched her hand as high into the air as it would go without her leaving her seat, but Harry didn’t have the faintest idea what a bezoar was. He tried not to look at Malfoy, Crabbe, and

Goyle, who were shaking with laughter.

“I don’t know, sir.”

“Thought you wouldn’t open a book before coming, eh, Potter?”

Harry forced himself to keep looking straight into those cold eyes. He had looked through his books at the Dursleys’, but did Snape expect him to remember everything in One Thousand Magical Herbs and Fungi?

Snape was still ignoring Hermione’s quivering hand.

“What is the difference, Potter, between monkshood and wolfs-bane?”

At this, Hermione stood up, her hand stretching toward the dungeon ceiling.

“I don’t know,” said Harry quietly. “I think Hermione does, though, why don’t you try her?”

A few people laughed; Harry caught Seamus’s eye, and Seamus winked. Snape, however, was not pleased.

“Sit down,” he snapped at Hermione. “For your information, Potter, asphodel and wormwood make a sleeping potion so powerful it is known as the Draught of Living Death. A bezoar is a stone taken from the stomach of a goat and it will save you from most poisons. As for monkshood and wolfsbane, they are the same plant, which also goes by the name of aconite. Well? Why aren’t you all copying that down?”

There was a sudden rummaging for quills and parchment. Over the noise, Snape said, “And a point will be taken from Gryffindor House for your cheek, Potter.”

Things didn’t improve for the Gryffindors as the Potions lesson continued. Snape put them all into pairs and set them to mixing up a simple potion to cure boils. He swept around in his long black cloak, watching them weigh dried nettles and crush snake fangs, criticizing almost everyone except Malfoy, whom he seemed to like. He was just telling everyone to look at the perfect way Malfoy had stewed his horned slugs when clouds of acid green smoke and a loud hissing filled the dungeon. Neville had somehow managed to melt Seamus’s cauldron into a twisted blob, and their potion was seeping across the stone floor, burning holes in people’s shoes.

Within seconds, the whole class was standing on their stools while Neville, who had been drenched in the potion when the cauldron collapsed, moaned in pain as angry red boils sprang up all over his

arms and legs.

“Idiot boy!” snarled Snape, clearing the spilled potion away with one wave of his wand. “I suppose you added the porcupine quills before taking the cauldron off the fire?”

Neville whimpered as boils started to pop up all over his nose.

“Take him up to the hospital wing,” Snape spat at Seamus. Then he rounded on Harry and Ron, who had been working next to Neville.

“You — Potter — why didn’t you tell him not to add the quills? Thought he’d make you look good if he got it wrong, did you? That’s another point you’ve lost for Gryffindor.”

This was so unfair that Harry opened his mouth to argue, but Ron kicked him behind their cauldron.

“Don’t push it,” he muttered, “I’ve heard Snape can turn very nasty.”

As they climbed the steps out of the dungeon an hour later, Harry’s mind was racing and his spirits were low. He’d lost two points for Gryffindor in his very first week — why did Snape hate him so much?

“Cheer up,” said Ron, “Snape’s always taking points off Fred and George. Can I come and meet Hagrid with you?”

At five to three they left the castle and made their way across the grounds. Hagrid lived in a small wooden house on the edge of the forbidden forest. A crossbow and a pair of galoshes were outside the front door.

When Harry knocked they heard a frantic scrabbling from inside and several booming barks. Then Hagrid’s voice rang out, saying, “ Back, Fang — back.”

Hagrid’s big, hairy face appeared in the crack as he pulled the door open.

“Hang on,” he said. “ Back, Fang.”

He let them in, struggling to keep a hold on the collar of an enormous black boarhound.

There was only one room inside. Hams and pheasants were hanging from the ceiling, a copper kettle was boiling on the open fire, and in the corner stood a massive bed with a patchwork quilt over it.

“Make yerselves at home,” said Hagrid, letting go of Fang, who bounded straight at Ron and started licking his ears. Like Hagrid, Fang was clearly not as fierce as he looked.

“This is Ron,” Harry told Hagrid, who was pouring boiling water into a large teapot and putting rock cakes onto a plate.

“Another Weasley, eh?” said Hagrid, glancing at Ron’s freckles.

“I spent half me life chasin’ yer twin brothers away from the forest.”

The rock cakes were shapeless lumps with raisins that almost broke their teeth, but Harry and Ron pretended to be enjoying them as they told Hagrid all about their first lessons. Fang rested his head on Harry’s knee and drooled all over his robes.

Harry and Ron were delighted to hear Hagrid call Filch “that old git’’

“An’ as fer that cat, Mrs. Norris, I’d like ter introduce her to Fang sometime. D’yeh know, every time I go up ter the school, she follows me everywhere. Can’t get rid of her — Filch puts her up to it.”

Harry told Hagrid about Snape’s lesson. Hagrid, like Ron, told Harry not to worry about it, that Snape liked hardly any of the students.

“But he seemed to really hate me.”

“Rubbish!” said Hagrid. “Why should he?”

Yet Harry couldn’t help thinking that Hagrid didn’t quite meet his eyes when he said that.

“How’s yer brother Charlie?” Hagrid asked Ron. “I liked him a lot — great with animals.”

Harry wondered if Hagrid had changed the subject on purpose.

While Ron told Hagrid all about Charlie’s work with dragons, Harry picked up a piece of paper that was lying on the table under the tea cozy. It was a cutting from the Daily Prophet:

GRINGOTTS BREAK-IN LATEST

Investigations continue into the break-in at Gringotts on 31 July, widely believed to be the work of Dark wizards or witches unknown. Gringotts goblins today insisted that nothing had been taken. The vault that was searched had in fact been emptied the same day. “But we’re not telling you what was in there, so keep your noses out if you know what’s good for you,” said a Gringotts spokesgoblin this afternoon.

Harry remembered Ron telling him on the train that someone had tried to rob Gringotts, but Ron hadn’t mentioned the date.

“Hagrid!” said Harry, “that Gringotts break-in happened on my birthday! It might’ve been happening while we were there!”

There was no doubt about it, Hagrid definitely didn’t meet Harry’s eyes this time. He grunted and offered him another rock cake. Harry read the story again. The vault that was searched had in fact been emptied earlier that same day. Hagrid had emptied vault seven hundred and thirteen, if you could call it emptying, taking out that grubby little package. Had that been what the thieves were looking for?

As Harry and Ron walked back to the castle for dinner, their pockets weighed down with rock cakes they’d been too polite to refuse, Harry thought that none of the lessons he’d had so far had given him as much to think about as tea with Hagrid. Had Hagrid collected that package just in time? Where was it now? And did Hagrid know something about Snape that he didn’t want to tell Harry?

Файл oaip\_file.txt

gfhghgjghfdgohofggffgfghgf

+

So she was considering in her own mind, as well as she could, for the hot day made her feel very sleepy and

stupid, whether the pleasure of making a , daisy-chain would be worth the trouble of getting up and picking the

daisies, when suddenly a White Rabbit with pink eyes ran close by her.

Файл code.in.txt

from PyQt5 import QtCore, QtWidgets, QtGui, QtWinExtras

import sys

class MyWindow(QtWidgets.QWidget):

def \_\_init\_\_(self, parent=None):

QtWidgets.QWidget.\_\_init\_\_(self, parent,

flags=QtCore.Qt.Window |

QtCore.Qt.MSWindowsFixedSizeDialogHint)

self.setWindowTitle("Класс QWinTaskbarButton")

# Создаем оба значка

self.icon1 = QtGui.QIcon("icon1.png")

self.icon2 = QtGui.QIcon("icon2.png")

# Создаем кнопку панели задач

self.taskbarButton = QtWinExtras.QWinTaskbarButton(parent=self)

# Создаем все три кнопки

hbox = QtWidgets.QHBoxLayout()

btnIcon1 = QtWidgets.QPushButton(self.icon1, "Значок &1")

btnIcon1.clicked.connect(self.setIcon1)

hbox.addWidget(btnIcon1)

btnIcon2 = QtWidgets.QPushButton(self.icon2, "Значок &2")

btnIcon2.clicked.connect(self.setIcon2)

hbox.addWidget(btnIcon2)

btnClearIcon = QtWidgets.QPushButton("&Убрать значок")

btnClearIcon.clicked.connect(self.taskbarButton.clearOverlayIcon)

hbox.addWidget(btnClearIcon)

self.setLayout(hbox)

self.resize(200, 50)

# После вывода окна на экран привязываем его к кнопке панели задач

def showEvent(self, evt):

self.taskbarButton.setWindow(self.windowHandle())

# Выполняем задание новых значков для кнопки панели задач при нажатии

# кнопок

def setIcon1(self):#sdfsdfsdfsdf

self.taskbarButton.setOverlayIcon(self.icon1)#fgdgfdg

def setIcon2(self): #jjfskdgsdfj

self.taskbarButton.setOverlayIcon(self.icon2) # fdsfsdfsd

#fddg df

#rgsghddgh

#fdgdsgfsdgsfdgd

#sfdgdhdfhfgjfhjhfj

app = QtWidgets.QApplication(sys.argv)#dfgdfhfhgf

window = MyWindow()

window.show()

sys.exit(app.exec\_())

Файл important.txt

vujgvmCfb tj ufscfu ouib z/vhm jdjuFyqm jt fscfuu uibo jdju/jnqm fTjnqm tj scfuuf ibou fy/dpnqm yDpnqmf jt cfuufs boui dbufe/dpnqmj uGmb tj fuufsc ouib oftufe/ bstfTq jt uufscf uibo otf/ef uzSfbebcjmj vout/dp djbmTqf dbtft (ubsfo djbmtqf hifopv up csfbl ifu t/svmf ipvhiBmu zqsbdujdbmju fbutc uz/qvsj Fsspst tipvme wfsof qbtt foumz/tjm omfttV mjdjumzfyq odfe/tjmf Jo fui dfgb pg hvjuz-bncj gvtfsf fui ubujpoufnq up ftt/hv Uifsf vmetip fc pof.. boe sbcmzqsfgf zpom pof pvt..pcwj xbz pu pe ju/ Bmuipvhi uibu bzx bzn puo cf wjpvtpc bu jstug ttvomf sfzpv( i/Evud xOp tj scfuuf ibou /ofwfs uipvhiBm fsofw jt fopgu cfuufs boui iu++sjh x/op gJ ifu nfoubujpojnqmf tj eibs pu mbjo-fyq tju( b bec /jefb Jg fui foubujpojnqmfn jt fbtz up bjo-fyqm ju znb cf b hppe jefb/ bnftqbdftO bsf pof ipoljoh sfbuh efbj .. fu(tm pe psfn gp tf"uip