# Лабораторная работа № 4

## «Основные характеристики качества и надежности программного продукта»

***Задание 1. Постановка задачи***

Разработать простую программу для управления списком задач (To-Do List) на Python. Программа должна:

1. Предложить пользователю добавить, редактировать, удалить или просмотреть введенные ранее задачи.
2. Выполнить выбранную операцию и отобразить результат пользователю.
3. Сохранять список задач между запусками программы.
4. Корректно обрабатывать неверные входные данные (например, некорректные номера задач, нечисловые значения) и исключительные ситуации (например, удаление несуществующей задачи).
5. Быть удобной в использовании, надежной и соответствовать стандартам качества программного обеспечения.

Программа будет оцениваться на основе ее функциональности, удобства использования, эффективности, ремонтопригодности, переносимости и надежности.

***Задание 2. Теоретические сведения***

Качество программного обеспечения – это совокупность характеристик, определяющих степень соответствия программного продукта требованиям заказчика, ожиданиям конечных пользователей, а также принятым стандартам разработки и безопасности. К основным характеристикам качества и надежности программного обеспечения относятся:

1. Функциональность: Определяет, насколько программное обеспечение соответствует заявленным требованиям и выполняет свои задачи без ошибок.
2. Надежность: Показывает, насколько стабильно и безотказно работает программное обеспечение в течение определенного времени и при различных условиях эксплуатации.
3. Удобство использования: Характеризует, насколько легко пользователи могут освоить, понять и эффективно работать с программой.
4. Эффективность: Определяет, насколько быстро и эффективно программное обеспечение выполняет свои задачи при разных нагрузках
5. Удобство в обслуживании: Простота, с которой программное обеспечение может быть изменено или обновлено.
6. Портативность: Определяет, насколько хорошо ПО взаимодействует с различными системами, средами и платформами.

***Задание 3. Обоснование качества программного средства***

Программа To-Do List придерживается следующих принципов качества и надежности программного обеспечения:

* 1. Функциональность:
  + Программа корректно выполняет четыре основные операции: добавление задачи, а также редактирование, удаление и просмотр созданных задач.
  + Она обрабатывает нестандартные ситуации, такие как удаление несуществующей задачи, путем выдачи соответствующего сообщения об ошибке.
  1. Надежность:
  + Программа включает обработку ошибок для управления неверными входными данными (например, некорректными номерами задач, нечисловыми значениями) и исключительными ситуациями (например, удаление несуществующей задачи).
  + Программа использует блоки "try-except" для перехвата и обработки исключений, гарантируя, что программа не выйдет из строя неожиданно.
  1. Удобство использования:
  + Программа предоставляет пользователю четкие инструкции, такие как запрос на ввод данных, описание вывода и список доступных операций.
  + Сообщения об ошибках являются наглядными и удобными для пользователя, помогая пользователям понять и исправить свои ошибки.
  1. Эффективность:
  + Программа выполняет вычисления с высокой скоростью и оптимальным использованием системных ресурсов за счёт минимизации излишних операций.
  + Программа не содержит алгоритмов высокой сложности, что обеспечивает оптимальную производительность.
  1. Удобство в обслуживании:
  + Код хорошо структурирован и легко читается, используются понятные имена переменных, позволяющие в нём ориентироваться.
  + Добавление новых функций (например, выставление приоритета задач) или изменение существующих потребует минимальных усилий.
  1. Поддерживаемость:
  + Программа написана на кроссплатформенном языке Python, что делает ее совместимой с различными операционными системами (Windows, macOS, Linux, Android)
  + Она не зависит от особенностей конкретной платформы, обеспечивая широкую совместимость.

***Задание 4. Текст программы***

В рамках лабораторной работы была разработана программа простого листа задач (To-Do List). Код программы представлен ниже:

import json

file\_tasks = "tasks.json"

def load\_tasks():

try:

with open(file\_tasks, "r", encoding="UTF-8") as file:

return json.load(file)

except Exception as e:

print(f"Ошибка: {e}")

return []

def save\_task(tasks):

try:

with open(file\_tasks, "w") as file:

json.dump(tasks, file, indent=4)

except ValueError as e:

print(f"Ошибка: {e}")

def add\_task(task):

tasks = load\_tasks()

tasks.append(task)

save\_task(tasks)

print("\nЗадача добавлена!")

def edit\_task(index, new\_task):

tasks = load\_tasks()

if 0 <= index <= len(tasks):

tasks[index] = new\_task

save\_task(tasks)

print("\nЗадача изменена!")

else:

print("\nНекорректный номер задачи!")

def delete\_task(index):

tasks = load\_tasks()

if 0 <= index < len(tasks):

tasks.pop(index)

save\_task(tasks)

print("\nЗадача удалена!")

else:

raise ValueError("\nНекорретный номер задачи!")

def list\_tasks():

tasks = load\_tasks()

if tasks:

print("\nСписок задач: ")

for i, task in enumerate(tasks):

print(f"{i+1}. {task}")

else:

print("Список задач пуст.")

def To\_Do\_List():

try:

while True:

print("\nМеню:")

print("1. Добавить задачу")

print("2. Редактировать задачу")

print("3. Удалить задачу")

print("4. Показать список задач")

print("5. Выйти")

choice = input("\nВыберите операцию: ")

if choice == "1":

task = input("\nВведите задачу: ")

add\_task(task)

elif choice == "2":

list\_tasks()

index = int(input("\nВведите номер задачи для редактирования: ")) - 1

new\_task = input("Введите новый текст задачи: ")

edit\_task(index, new\_task)

elif choice == "3":

list\_tasks()

index = int(input("Введите номер задачи для удаления: ")) - 1

delete\_task(index)

elif choice == "4":

list\_tasks()

elif choice == "5":

print("Выход из программы.")

break

else:

print("Некорректный операция, попробуйте снова!")

except ValueError as e:

print(f"Ошибка: {e}")

except Exception as e:

print(f"Произошла непредвиденная ошибка: {e}")

To\_Do\_List()

Программа выполняет основные операции управления списком задач, такие как добавление, удаление, редактирование и просмотр задач, сохраняя данные в файле для последующего использования.

***Задание 5. Контрольный пример***

Рассмотрим контрольные примеры работы программы в таблице 1.

Таблица 1 - результаты тестирования программы

|  |  |  |  |
| --- | --- | --- | --- |
| Тестовый пример | Вводные данные | Ожидаемый результат | Оценка качества |
| Нормальная работа, добавление задачи | Купить молоко | Задача добавлена! | Функциональность, удобство использования, надежность |
| Нормальная работа, редактирование задачи | 1 → Купить хлеб | Задача изменена! | Функциональность, удобство использования, надежность |
| Нормальная работа, просмотр задач | - | 1. Купить молоко | Функциональность, удобство использования, надежность |
| Нормальная работа, удаление задачи | 1 | Задача удалена! | Функциональность, удобство использования, надежность |
| Удаление несуществующей задачи | 2 | Некорректный номер задачи! | Функциональность, удобство использования, надежность |
| Некорректный ввод | abc | Ошибка: invalid literal for int() with base 10: 'abc' | Функциональность, удобство использования, надежность |
| Недопустимый символ операции | 6 | Некорректный операция, попробуйте снова! | Функциональность, удобство использования, надежность |

Результаты тестирования программы показали ожидаемый результат во всех случаях, система является функциональной, удобной в использовании и надежной

***Задание 6. Выводы о проделанной работе***

Программа To-Do List демонстрирует высокие показатели качества и надежности программного обеспечения. Она успешно реализует основные критерии функциональности, удобства использования, эффективности, ремонтопригодности, переносимости и надежности. Благодаря соблюдению этих принципов программа представляет собой надежный и удобный инструмент для эффективного управления списком задач. В дальнейшем возможны улучшения, такие как добавление приоритетов задач, установки сроков выполнения, а также разработка графического интерфейса, что сделает программу еще более удобной и функциональной.