**Princess Sumaya University for Technology**

King Abdullah II Faculty of Engineering

Electrical Engineering Department



**Microprocessors (22344)**

**Software Quiz App Assembly Project**

**Author:** Qais Jildeh (20210155)

**Supervisor:** Dr. Heba Abdel-Nabi

*August 6, 2024*

***Abstract***

*This project deploys a variable Multiple-Choice Quiz Application in 8086 assembly language with user login, randomly sorted questions, and scoring. Users are verified through preassigned credentials, respond dynamically requested questions on the selected subject matter, and get instant feedback. The solution focuses on modular design, memory management, and interrupt-based I/O, demonstrating deployed low-level programming to interactive software.*

**Table of Contents**

[Introduction 1](#_Toc197895544)

[Requirements 1](#_Toc197895545)

[Procedure 1](#_Toc197895546)

[Results and Discussion 2](#_Toc197895547)

[Conclusion 2](#_Toc197895548)

[References 2](#_Toc197895549)

No table of figures entries found.

## Introduction

This project creates an 8086-assembly language customizable quiz game with user authentication, dynamic question shuffling, and scoring. Based on modular procedures as its basic structure, it exhibits effective memory management and input handling in real-time, demonstrating real-world uses of low-level programming for interactive systems.

### Requirements

* User Authentication: Develop a login system with predefined credentials and a 3-attempt limit.
* Dynamic Question Handling: Load, shuffle, and display 4 MCQs on a user-chosen topic.
* Scoring System: Calculate and display user scores.
* Modular Design: Organize code into reusable procedures and analyze performance efficiency.

## Procedure

The quiz a

## Results and Discussion

dsafasf

## Conclusion

This project implemented an MCQ Quiz Game in 8086 assembly with customizable functionality, with successful key features: a safe 3-attempts login mechanism based on family-based credentials, dynamic shuffling of 4 Cars-themed questions, case-insensitive answer checking, and real-time score update with feedback. Modular code structure was optimized for efficiency, with greater emphasis placed on hardware interaction and resource handling, while demonstrating real-world applications of low-level programming in designing interactive systems.

## References

<https://www.tutorialspoint.com/microprocessor/microprocessor_8086_instruction_sets.htm>

<https://yassinebridi.github.io/asm-docs/8086_bios_and_dos_interrupts.html#int21h>

<https://yassinebridi.github.io/asm-docs/8086_bios_and_dos_interrupts.html>