**Computers and control systems engineering department**

**Faculty of engineering**

**Mansoura University**

**Operating Systems Project 2022/2023**

Guessing Game Script

**By: Seif El-Din Wael Nabil Sweilam**

**ID: 1000234090**

**Supervisors:**

**Dr: Asmaa Hamdy**

**Eng: Menna Sedik**

**Table of Contents**

* **Abstract**
* **Introduction**
* **Design, Implementation and Results**

**(screenshots from Your Project)**

* **Conclusion**
* **Reference**

**Abstraction & Introduction**

This project is a number guessing game designed in the way that the user thinks of any number between 1 and 1000, and the script will try to get the correct guess as fast as possible. The script is written in bash language. The mechanism of this game applies the concept of binary search, where we set a beginning point, an end point and compare the value in the middle, according to the result we adjust the beginning and end point to omit half of the interval. This process is continuous until we find the correct guess.

The script can run on Linux devices inside the terminal, and on windows devices through bash-like third-party applications as Git Bash, Cmder.

**Design**

The script contains a main function called “play” which is responsible for executing the whole game. It sets the start point to 1 and the end point to 1000 to cover the whole interval, then it introduces the game to the user and gives them 10 seconds to think of a number. The program starts to guess using the midpoint of the start and end points. The guessing is continuous where each time the user is prompted to decide whether the guess is correct, higher or lower than the answer. The response of the user is handled in a case statement which modifies the start and the end according to the response. If the user enters an invalid input, they will be prompted to reenter a valid one.

When the program guesses the correct answer, it will ask the user if they would like to play again. This will recall the “play” method one more time repeating the whole logic.

**Implementation**

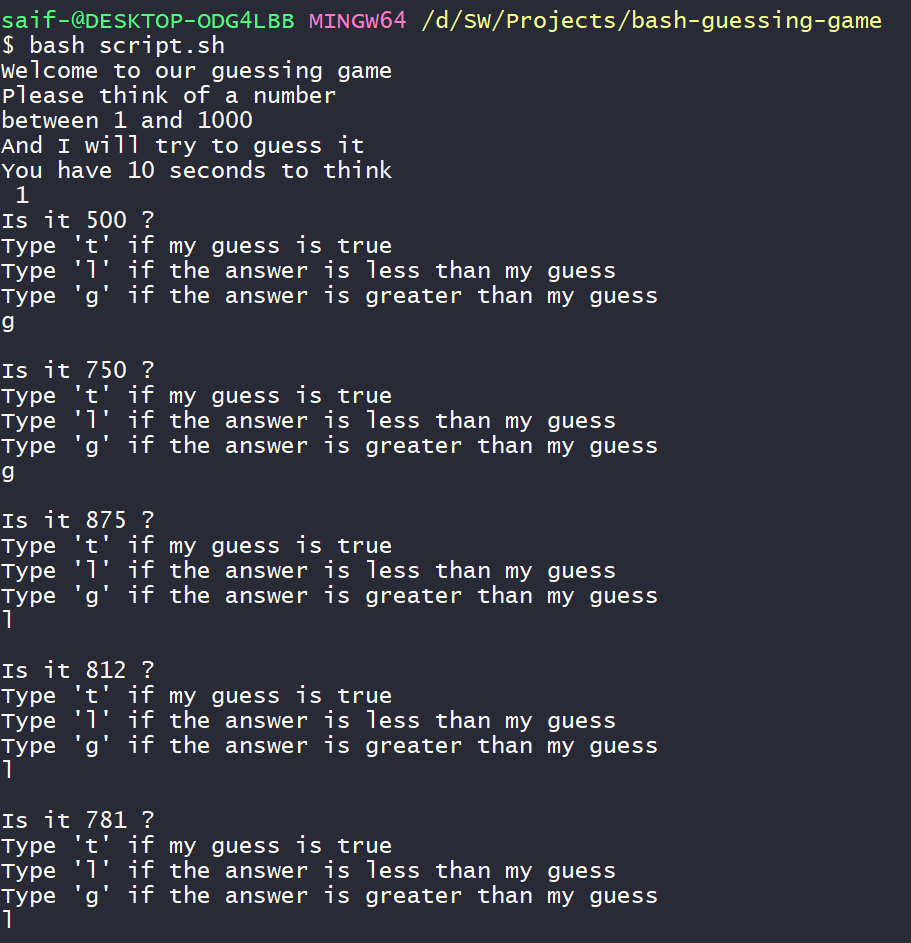
1. Open the terminal and navigate to the working directory
2. Enter the command: nano script.sh
3. Write the following code:

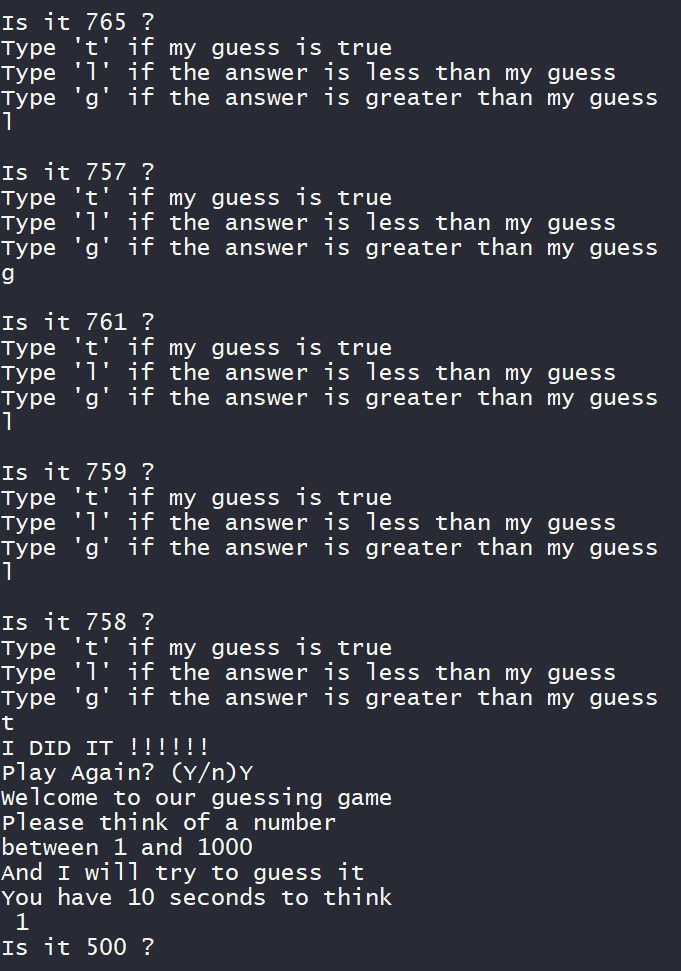
Text

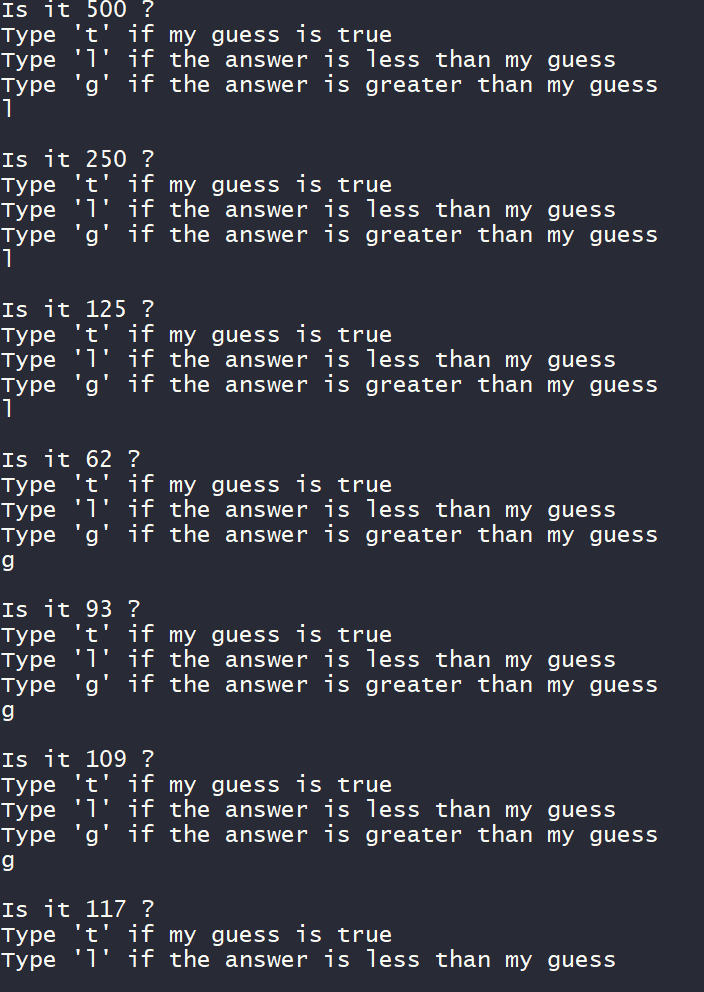
Description automatically generated

1. Save the file (Ctrl + S) and Close nano (Ctrl + K)
2. Enter command: bash script.sh

**Result**







Text

Description automatically generated

**Conclusion**

In this project I made use of what I learned in Linux and bash scripting, and what I learned in functional programming to implement the logic of the game.

**References**

* [bash script, erase previous line? - Stackoverflow](https://stackoverflow.com/questions/5861428/bash-script-erase-previous-line)

https://stackoverflow.com/questions/5861428/bash-script-erase-previous-line

* [How to Pause a Bash Script | Linux Sleep Command - Its Linux FOSS](https://itslinuxfoss.com/linux-sleep-command-pause-bash-script/#:~:text=The%20sleep%20command%20adds%20the%20delay%20time%20in,%28s%29%2C%20minutes%20%28m%29%2C%20hours%20%28h%29%2C%20and%20days%20%28d%29.)

https://itslinuxfoss.com/linux-sleep-command-pause-bash-script/

* [Bash Functions - Linuxize](https://linuxize.com/post/bash-functions/)

https://linuxize.com/post/bash-functions/

* [Bash case statement [Explained with 3 examples] - jquery-az.com](https://www.jquery-az.com/bash-case/#:~:text=In%20Bash%2C%20the%20Case%20statement%20is%20one%20of,ask%20the%20user%20to%20enter%20a%20color%20code.)

https://www.jquery-az.com/bash-case/

* [Bash IF – Syntax and Examples - TutorialKart](https://www.tutorialkart.com/bash-shell-scripting/bash-if/)

https://www.tutorialkart.com/bash-shell-scripting/bash-if/