

NUMBER GUESSING GAME

A Course Based Project Report Submitted in partial fulfillment of
the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY IN CSE (CYBER SECURITY)

Submitted by

R.ARPITHA

21071A6247

R.VENU GOPAL REDDY

21071A6248



DEPARTMENT OF CSE (CYBER SECURITY)

**VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING
AND TECHNOLOGY**

An Autonomous Institute, NAAC Accredited with 'A++' Grade NBA Accredited for CE, EEE, ME, ECE,
CSE, EIE, IT B. Tech Courses Approved by AICTE, New Delhi,
Affiliated to JNTUH Recognized as "College with Potential for Excellence" by UGC ISO 9001:2015
Certified, QS I GUAGE Diamond Rated

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade NBA Accredited for
CE, EEE, ME, ECE, CSE, EIE, IT B. Tech Courses Approved by AICTE, New Delhi,
Affiliated to JNTUH Recognized as "College with Potential for Excellence" by
UGC ISO 9001:2015 Certified, QS I GUAGE Diamond Rated

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

DEPARTMENT OF CSE (CYBER SECURITY)



CERTIFICATE

This is to certify that the project report entitled "NUMBER GUESSING" is a bonafide work done under our supervision and is being submitted by **R.ARPITHA(21071A6247),R.VENU GOPAL(21071A6248)** in partial fulfillment for the award of the degree of Bachelor of Technology in CSE (CYBER SECURITY) of the VNRVJIET, Hyderabad during the academic year 2022-2023.
Certified further that to the best of our knowledge the work presented in this thesis has not been submitted to any other University or Institute for the award of any Degree or Diploma.

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade NBA Accredited for CE,
EEE, ME, ECE, CSE, EIE, IT B. Tech Courses Approved by AICTE, New Delhi,
Affiliated to JNTUH Recognized as "College with Potential for Excellence" by
UGC ISO 9001:2015 Certified, QS I GUAGE Diamond Rated

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

DEPARTMENT OF CSE (CYS)



DECLARATION

We declare that the major project work entitled "number guessing" submitted department of CSE(CYS), Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **CSE(CYS)** is a bonafide record of our own work carried out under the supervision of **P.LALITHA, Assistant Professor**. Also, we declare that the matter embodied in this thesis has not been submitted by us in full or in any part thereof for the award of any degree/diploma of any other institution or university previously.

Place: Hyderabad

ABSTRACT:

- > Build a Number guessing game, in which the user selects a range.
- > Let's say User selected a range, i.e., from A to B, where A and B belong to Integer.
- > Some random integer will be selected by the system and the user has to guess that integer in the minimum number of guesses
- > The number Guessing game in Python is a common project for programmers who have recently learned a new language and mastered topics like number generation and conditional statements with iteration.

Table of Contents	Page No
1. INTRODUCTION.....	11
2. LITERATURE SURVEY.....	12
3. DESIGN.....	13
3.1. REQUIREMENT SPECIFICATION (S/W & H/W)-----	13
3 2. UML DIAGRAMS.....	14
4. IMPLEMENTATION.....	15
5. TESTING.....	18
5 1. TEST CASES.....	18
5 2. TEST RESULTS.....	19
6. RESULTS.....	20
7. CONCLUSION.....	20
8. FUTURE SCOPE.....	21
BIBLIOGRAPHY.....	21

1.INTRODUCTION

-> The foundation of the number guessing game in Python is the player's assumption that they can guess a number within the given range. The player wins the game if they correctly guess the target number; else, they lose. The player must specify how many of the game's restricted attempts they have since it has limited efforts. The player will lose the game if this happens.

-> There are a few tasks that we need to take care of while building a number-guessing game using Python. First of all, we need the user to select a range of numbers in which the number-guessing game will be played. Once the user has selected the range from X to Y, where X and Y are integers. Some random number will be selected by the integer and the user has to guess that integer in the minimum number of guesses.

So, How do we find this minimum number of guesses? Well, that minimum number of guesses depends upon range and we already have the formula for it.

2. LITERATURE SURVEY

Game Concepts:

As game and the SDL library are portable across different platforms and devices, they both need to define and work with abstractions for various hardware realities. Understanding those concepts and abstractions will help you design and develop your own games.

Initialization and Modules:

The game library is composed of a number of Python constructs, which include several different modules.

These modules provide abstract access to specific hardware on your system, as well as uniform methods to work with that hardware. For example, `display` allows uniform access to your video display, while `joystick` allows abstract control of your joystick.

After importing the game library in the example above, the first thing you did was initialize Game using `game.init()`. This function calls the separate `init()` functions of all the included game modules. Since these modules are abstractions for specific hardware, this initialization step is required so that you can work with the same code on Linux, Windows, and Mac.

Displays and Surfaces:

In addition to the modules, pygame also includes several Python classes, which encapsulate non-hardware dependent concepts. One of these is the `Surface` which, at its most basic, defines a rectangular area on which you can draw. `Surface` objects are used in many contexts in pygame. Later you'll see how to load an image into a `Surface` and display it on the screen.

In pygame, everything is viewed on a single user-created display, which can be a window or a full screen. The display is created using `.set_mode()`, which returns a `Surface` representing the visible part of the window. It is this `Surface` that you pass into drawing functions like `pygame.draw.circle()`, and the contents of that `Surface` are pushed to the display when you call `game.display.flip()`.

Images and Rects:

Your basic game program drew a shape directly onto the display's `Surface`, but you can also work with images on the disk. The `image` module allows you to load and save images in a variety of popular formats.

Images are loaded into `Surface` objects, which can then be manipulated and displayed in numerous ways.

As mentioned above, `Surface` objects are represented by rectangles, as are many other objects in pygame, such as images and windows. Rectangles are so heavily used that there is a special `Rect` class just to handle them. You'll be using `Rect` objects and images in your game to draw players and enemies, and to manage collisions between them.

3. DESIGN

3.1 REQUIREMENT SPECIFICATION

Number guessing game is a simple game which is suitable for different operating systems like Windows,

Mac, and Linux. It is a user-friendly application.

Software Requirements:

IDLE (Python 3.11 64-bit)

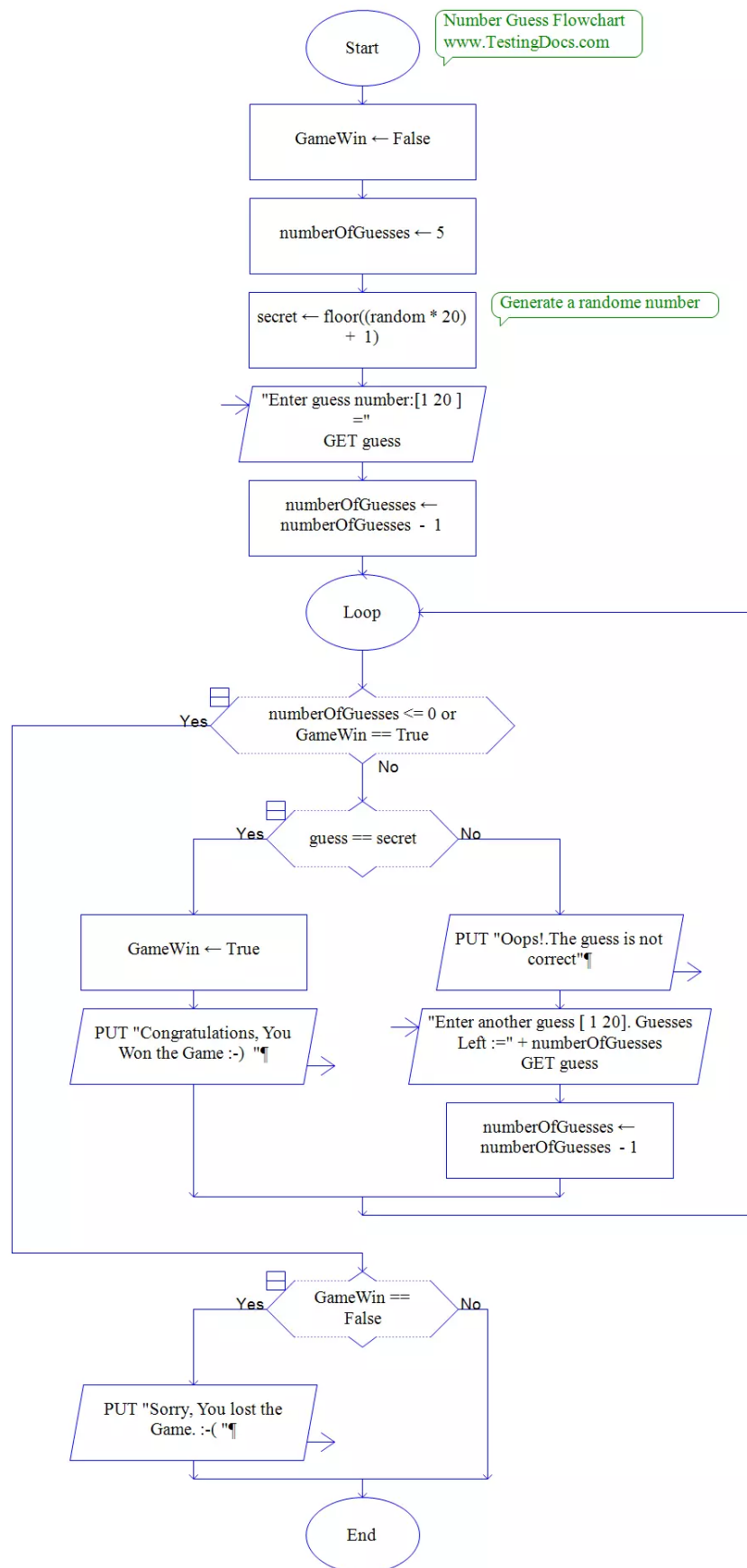
Hardware Requirements:

Operating System: Windows 7

Processor: Intel Core i3

Disc Space: 256MB

3.2UML Diagram: ACTIVITY DIAGRAM:



4.IMPLEMENTATION

CODE:

```
import random
import math
# Taking Inputs
lower = int(input("Enter Lower bound:- "))

# Taking Inputs
upper = int(input("Enter Upper bound:- "))

# generating random number between
# the lower and upper
x = random.randint(lower, upper)
print("\n\tYou've only ",
      round(math.log(upper - lower + 1, 2)),
      " chances to guess the integer!\n")

# Initializing the number of guesses.
count = 0

# for calculation of minimum number of
# guesses depends upon range
while count < math.log(upper - lower + 1, 2):
    count += 1

# taking guessing number as input
guess = int(input("Guess a number:- "))

# Condition testing
if x == guess:
    print("Congratulations you did it in ",
          count, " try")
    # Once guessed, loop will break
    break
elif x > guess:
    print("You guessed too small!")
elif x < guess:
    print("You Guessed too high!")

# If Guessing is more than required guesses,
# shows this output.
if count >= math.log(upper - lower + 1, 2):
    print("\nThe number is %d" % x)
    print("\tBetter Luck Next time!")

# Better to use This source Code on pycharm!
```

OUTPUT:

```
Enter Lower bound:- 1
Enter Upper Bound: 100

    You've only 7 chances to guess the integer!

Guess a number:- 50
You guessed too small!
Guess a number:- 75
You Guessed too high!
Guess a number:- 62
You Guessed too high!
Guess a number:- 56
You Guessed too high!
Guess a number:- 53
You guessed too small!
Guess a number:- 54
Congratulations you did it in 6 try

Process finished with exit code 0
|
```

		2	3			6	7		
10	11			14	15			18	19
		22	23			26	27		
30	31			34	35			38	39
		42	43			46	47		
50	51			54	55			58	59
		62	63			66	67		
70	71			74	75			78	79
		82	83			86	87		
90	91			94	95			98	99

5.RESULTS

The player can stop the game by clicking the x of the tab,if he chooses to resume he should to click the START or if he chooses to leave he need to click EXIT.at the end of the game the total score of the player is displayed.

6.FUTURE SCOPE

Artificial Intelligence: The future of this programming language can also be predicted by how it has helped and still helps AI technology.

Several Python frameworks, libraries, and tools are mainly developed to direct AI to overcome human efforts with enhanced efficiency for various development purposes.

Big Data: Apart from AI technology, the programming language has been successfully contributing in the domain of Big Data to analyze a large number of data sets with the help of its high-performance toolkits and libraries.

Networking: In networking, Python can be utilized for several purposes, such as reading, writing, and configuring routers and switches and performing several other networking automation tasks that are too cost-effective and secure, indicating that it holds a brighter scope and a brighter scope.

REFERENCES

<https://www.python.org>

<https://www.codingninjas.com>

<https://www.geeksforgeeks.org>