The Temple of Treasure Hunt

Bin Han

2025-07-16

Table of contents

1 Project Description

Temple of the Treasure Hunt is a thrilling console-based game where the player takes on the role of a daring treasure hunter exploring an ancient underground temple in search of a legendary treasure. Hidden behind one of ten massive doors, only one leads to the riches—while the rest offer nothing but dust and disappointment.

The challenge lies in finding the treasure with as few attempts as possible. For every incorrect door the player opens, the value of the treasure is cut in half. Success requires strategy, courage, and perhaps a bit of luck. The fewer doors opened, the greater the reward—and a place on the secret leaderboard of victorious explorers.

The game includes features to:

Start a new game and search for the treasure.

Save the player's name and number of attempts to a file (skattresultat.csv).

Display a leaderboard sorted by best performance (fewest attempts).

Show a bar chart illustrating how many players needed a given number of tries.

Temple of the Treasure Hunt blends luck, decision-making, and data visualization, providing both entertainment and a practical learning experience in programming, file handling, and basic statistics.

2 Loading Library

This is a Python code chunk:

```
import csv
import os
import random
import matplotlib.pyplot as plt
```

3 Define the file and the Path.

```
import os

# Use the current working directory instead of __file__
script_mapp = os.getcwd()  # Gets the current working directory
filnamn_input = "exempelCSV.csv"  # Replace with your actual filename or input method
filnamn = os.path.join(script_mapp, filnamn_input)

print('Du valde filnamnet', filnamn)
```

```
Du valde filnamnet C:\Users\xhanbi\Desktop\Kurs25\Python_Borås\Uppgift_1\exempelCSV.csv
```

4 Create the file with header if not exist.

```
#Create file it the file doesn't exist
def skapa_fil_om_den_saknas():
   if not os.path.exists(filnamn):
       with open (filnamn, "w", newline="") as fil:
            csv.writer(fil). writerow(["Namn", "Försök"])
```

5 Game:the treasure hunt

```
#Menu 1:Compute the name and number of the try
def SpelaSkattjaktan():
   treasure_door=random.randint(1,10)# random door with the treasure
   door_list=[]
   Trials=0

   print("\n--- Skattjakten---")
   while True:
      try:
        guessing = int(input("Door number from 1 to 10"))
        if guessing <1 or guessing >10:
          print(f"Invalid input, try again!")
          continue# Jump to the next while loop
        if guessing in door_list:
          print(f"door aleady opened!")
          continue
        door_list.append(guessing)
        Trials+=1
        if guessing ==treasure_door:
          print(f"Congratualtions! You have found the treaure behind the door {guessing} a
          name=input("Write down your name:")
          with open(filnamn, "a", newline="") as fil:
            csv.writer(fil).writerow([name,Trials])
            break

      except ValueError:#Other Input
        print("Invalid input, try again!")
```

6 Leaderboard

```
def visa_toplistan():
  print("---Topplista---")
  try:
    lines=[]
    with open(filnamn, "r", encoding="latin1") as data:
      reader=csv.reader(data)# iterable object
      header=next(reader)#skip the header
      for line in reader:
        if len(line)>=2 and line[1]. isdigit():
          lines.append(line)
```

```
    lines_sorted=sorted(lines, key=lambda x: int(x[1])) #Sorted lines by the second column
    print(f"{header[0]:<15} {header[1]:>7}")
    for line in lines_sorted:
      print(f"{line[0]:<15} {line[1]:>7}")
  except FileNotFoundError:
    print("The file does not exist yet. Play first.")

visa_toplistan()
```

```
---Topplista---
Namn            Försök
Spelare_3            1
Spelare_8            1
Spelare_40           1
Spelare_46           1
Spelare_2            2
Spelare_11           2
Spelare_23           2
Spelare_30           2
Spelare_35           2
Spelare_17           3
Spelare_36           3
Spelare_42           3
Spelare_50           3
Spelare_1            4
Spelare_10           4
Spelare_28           4
Spelare_6            5
Spelare_18           5
Spelare_26           5
Spelare_38           5
Spelare_43           5
Spelare_4            6
Spelare_19           6
Spelare_21           6
Spelare_25           6
Spelare_29           6
Spelare_37           6
Spelare_41           6
Spelare_44           6
Spelare_49           6
Spelare_5            7
Spelare_7            7
Spelare_22           7
Spelare_31           7
```

```
Spelare_45            7
Spelare_15            8
Spelare_20            8
Spelare_34            8
Spelare_47            8
Spelare_9             9
Spelare_13            9
Spelare_14            9
Spelare_24            9
Spelare_12           10
Spelare_16           10
Spelare_27           10
Spelare_32           10
Spelare_33           10
Spelare_39           10
Spelare_48           10
```

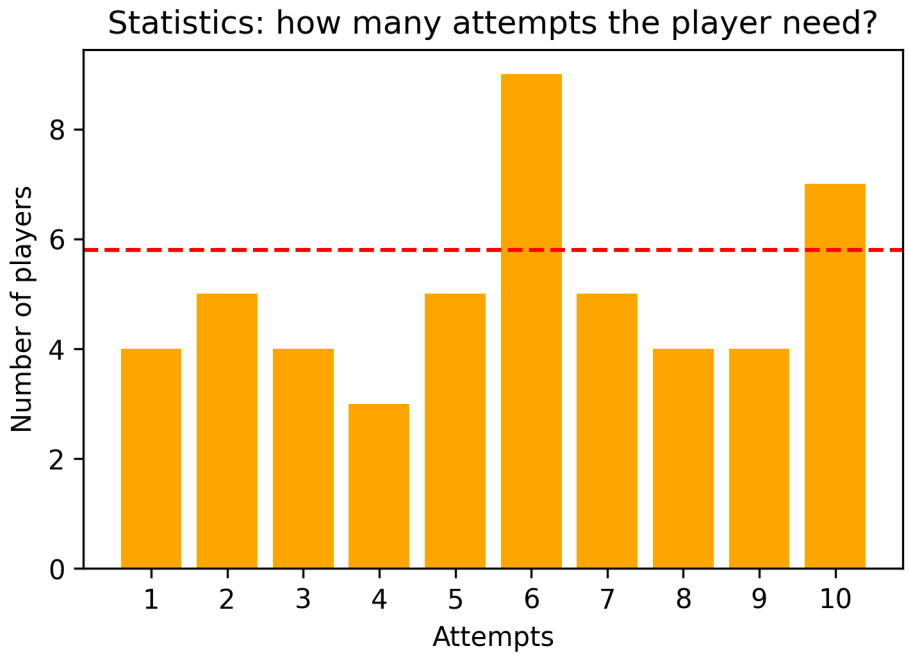
7 Chart of the result

```
def visa_diagram():
  print("---diagram of attempts---")
  try:
    försök_list=[]
    with open(filnamn, "r", encoding="latin1") as data:
      reader=csv.reader(data)
      next(reader)
      for rad in reader:
        if len(rad)>=2 and rad[1].isdigit():
         försök_list.append(int(rad[1]))
    försök_counts={}
    for item in försök_list:
      if item in försök_counts:
        försök_counts[item]+=1
      else:
        försök_counts[item]=1
    x=sorted(försök_counts.keys())
    y=[försök_counts[k] for k in x]
    print(x)
    print(y)
    allsum=0
    for i in range(len(x)):
      allsum+=x[i]*y[i]
    mean=allsum/sum(y)
```

```
# create te bar chart
    plt.bar(x, y, color="orange")
    plt.title("Statistics: how many attempts the player need?")
    plt.axhline(y=mean, color='red', linestyle='--', label=f'Mean = {mean:.2f}')
    plt.xlabel("Attempts")
    plt.ylabel("Number of players")
    plt.xticks(x)
    plt.show()
  except FileNotFoundError:
    print("The file does not exist yet. Play first.")

visa_diagram()
```

```
---diagram of attempts---
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
[4, 5, 4, 3, 5, 9, 5, 4, 4, 7]
```



From the bar plot, we can see that the player need 6 times to accomplish the game ranked first, ten times ranked seond. which are higher than the average attempts needed for this game.

8 End the game

```
def avsluta_spelet():
    print(f"Tack för att du deltagit och din resultat är sparade i din resultatfil.")
    exit()
```

9 Menu

```
def meny():
    skapa_fil_om_den_saknas()
    while True:
        print("\n--- Startmeny ---")
        print("1. Spela Skattjakten")
        print("2. Visa topplistan")
        print("3. Visa diagram över tidgare resultat")
        print("4. Avsluta")
        #val=input("valj ett alternativ (1-4): ")
        if val =="1":
            SpelaSkattjakten()
        elif val=="2":
            visa_toplistan()
        elif val=="3":
            visa_diagram()
        elif val=="4":
            avsluta_spelet()
        else:
            print("ogiltigt val. Försök igen.")


#if __name__== "__main__":
#    meny()
```