TASK 1:

Using Genetic Algorithms to guess a password given the number of correct letters in the guess. Build a mutation engine

CODE:

```python
# Owned
__author__ = "Qaiser Abbas"
__copyright__ = "Copyright 2020, Artificial Intelligence lab-10"
__email__ = "qaiserabbas889@yahoo.com"
#===========================================================
# {code}
import random
import datetime
import sys
import time
geneSet = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ!. '
password = "Genetic Algorithm by Qaiser!"
def generate_parent(length):
    gene = []
    while len(gene) < length:
        sample_size = min(length - len(gene), len(geneSet))
        # appends list with values ==> ['X','B'].extend('a','y') -
> ['X','B','a','y']
        gene.extend(random.sample(geneSet, sample_size))
    # converts list to str ==> ['a', 'b', ' ', 'C', 'N', 'x'] -> ab CNx
    return ''.join(gene)
def get_fitness(guess):
    return sum(1 for expected, actual in zip(password, guess) if expected == actu
al)
def mutate(parent):
    index = random.randrange(0, len(parent))
    child_genes = list(parent)
    new_gene, alternate = random.sample(geneSet, 2)
    child_genes[index] = alternate if new_gene == child_genes[index] else new_gen
e
    return ''.join(child_genes)
def display(guess):
    time_diff = datetime.datetime.now() - startTime
    fitness = get_fitness(guess)
    # print('{0}\t{1}\t{2}'.format(guess, fitness, time_diff))
    time.sleep(.001)
```

```python
    sys.stdout.write('\rGeneration #' + '\t' + guess)
if __name__ == '__main__':
    random.seed(4)
    startTime = datetime.datetime.now()
    best_parent = generate_parent(len(password))
    best_fitness = get_fitness(best_parent)
    display(best_parent)
    while True:
        child = mutate(best_parent)
        child_fitness = get_fitness(child)
        display(child)
        if best_fitness >= child_fitness:
            continue
        if child_fitness >= len(best_parent):
            break
        best_fitness = child_fitness
        best_parent = child
```

OUTPUT:

Generating Password:

```
PS C:\Users\iQais> & C:/Users/iQais/AppData/Local/Programs/Python/Python39
10/Lab-10-task-1.py"
Generation #     GtnUzEw AbgJsdoHI rl!QniAuSi
```

```
PS C:\Users\iQais> & C:/Users/iQais/AppData/Local/Programs/Pyt
10/Lab-10-task-1.py"
Generation #     GtnUzEc AbgJsdthm rl!QniserC
```

```
PS C:\Users\iQais> & C:/Users/iQais/AppData/Local/Programs/P
10/Lab-10-task-1.py"
Generation #     GenezEc Abgorithm by!QQiser!
```

Password Generated:

```
PS C:\Users\iQais> & C:/Users/iQais/AppData/Local/Programs/Python/Python3
10/Lab-10-task-1.py"
Generation #    Genetic Algorithm by Qaiser!
PS C:\Users\iQais> []
```