Tishreen University

Communication and electrical

Engineering

Network programming 1

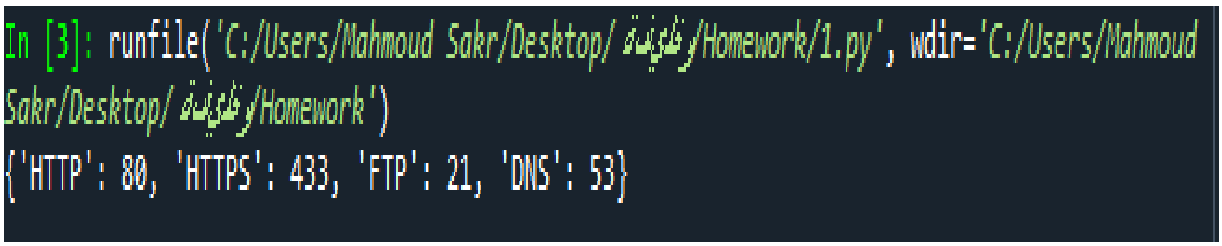
Name : Bassam Faisal Salman

Number : 2036

Qustion 1 :

#A

```
d= { }
L1 = ['HTTP','HTTPS','FTP','DNS']
L2 = [80,433,21,53]
for i,j in zip(L1,L2):
    d[i]=j
print(d)
```



#B

```
def factorial(n):
    if n == 0:
        return 1
```

```
    else:
        return n * factorial(n-1)

num = int(input("Enter a number: "))

if num < 0:
    print("Factorial is not defined for negative numbers")
elif num == 0:
    print("Factorial of 0 is 1")
else:
    print("Factorial of", num, "is", factorial(num))
```

```
In [1]: runfile('C:/Users/Mahmoud Sakr/Desktop/وظيفة بيثون/Homework/2.py', wdir='C:/Users/Mahmoud
Sakr/Desktop/وظيفة بيثون/Homework')
Enter a number: 15
Factorial of 15 is 1307674368000
```

#C

```
L= ['Network', 'Bio','Programming','Physics','Music']
i = 0
for i in range(len(L)):
```

```
    if L[i].startswith("B"):
        print(L[i])
```

```
In [5]: runfile('C:/Users/Mahmoud Sakr/Desktop/وظيفتي/Homework/3.py', wdir='C:/Users/Mahmoud
Sakr/Desktop/وظيفتي/Homework')
Bio
```

#D

```
d= {a:a+1 for a in range(0,11)}
print(d)
```

```
In [6]: runfile('C:/Users/Mahmoud Sakr/Desktop/وظيفتي/Homework/4.py', wdir='C:/Users/Mahmoud
Sakr/Desktop/وظيفتي/Homework')
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
```

Qustion 2 :

```
b_num = list(input("Input a binary number: "))
value = 0

for i in range(len(b_num)):
```

```
        digit = b_num.pop()
        if digit == '1':
            value = value + pow(2, i)
print("The decimal value of the number is", value)
```

```
In [7]: runfile('C:/Users/Mahmoud Sakr/Desktop/وظيفة/Homework/5.py', wdir='C:/Users/Mahmoud
Sakr/Desktop/وظيفة/Homework')
Input a binary number: 1011
The decimal value of the number is 11
```

Qustion 3:

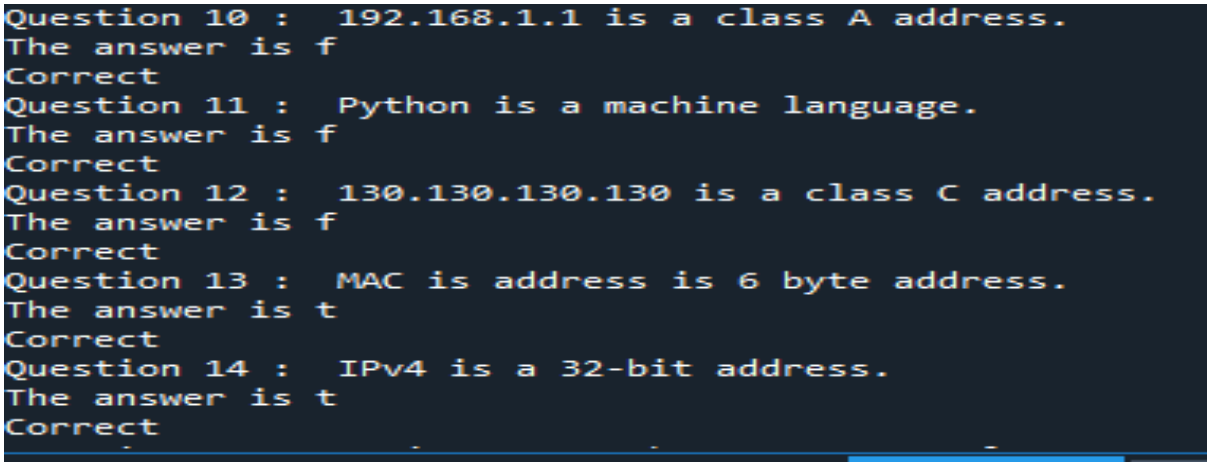
```
import json
questions = { }
#define a variable for the score
scores = 0
#define the question number
number=1
#loading question to the program
```

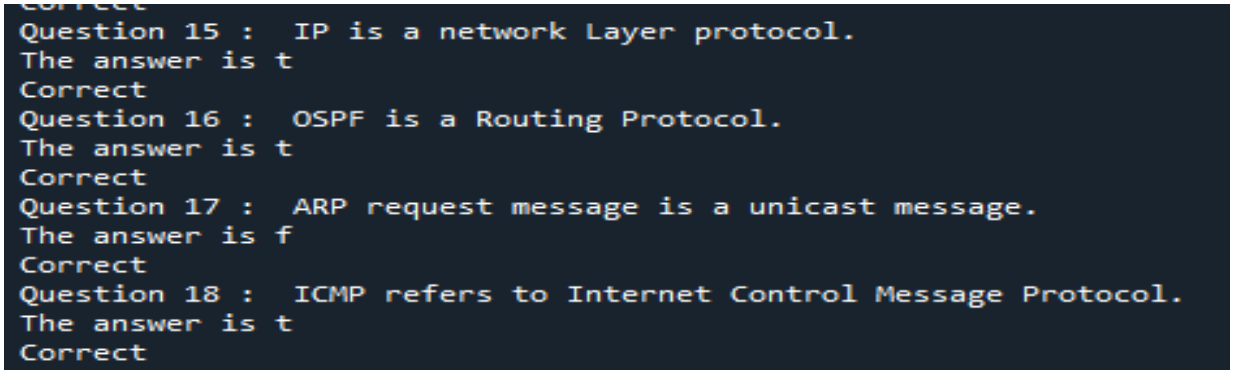
```
f = open("questions.txt",'r')
questions = json.load(f)
f.close()

print("python quiz programm")
print("Enter t for True or f for False")
name = input("Enter your full name: ")
#display the questions
for ques in questions.keys():
    #displaying the question
    print("Question",number,": ", ques)
    ans = input("The answer is ")
    #testing the result
    if ans.upper() == questions[ques].upper():
        scores = scores + 1
        print("Correct ")
    else:
        print ("Wrong")
    number = number + 1
```

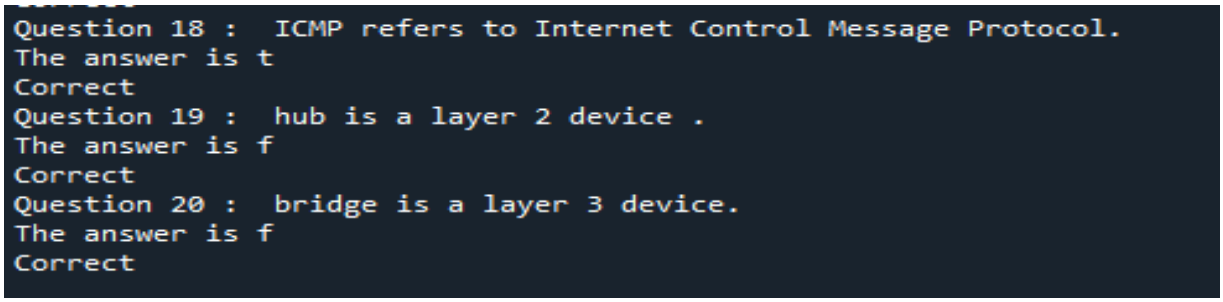
```
#write the name and the score is a separate file
result={name:scores}
m = open("score.txt",'w')
result = json.dump(result,m)
m.close()
```

```
In [9]: runfile('C:/Users/Mahmoud Sakr/Desktop/وظيفة/Homework/6.py', wdir='C:/Users/Mahmoud
Sakr/Desktop/وظيفة/Homework')
python quiz programm
Enter t for True or f for False
Enter your full name: Bassam faisal salman
Question 1 :  10.0.0.5 is a private ip address.
The answer is t
Correct
Question 2 :  153.16.2.8 is a private ip address.
The answer is f
Correct
Question 3 :  ARP refers to Address Resolution Protocol.
The answer is t
Correct
Question 4 :  TCP is a network layer protocol.
```

```
Question 5 :  IPv4 is a 128-bit address.
The answer is f
Correct
Question 6 :  IPv6 is a 128-bit address.
The answer is t
Correct
Question 7 :  SDN refers to Software Defined Network.
The answer is t
Correct
Question 8 :  UDP is a Transport Layer protocol.
The answer is t
Correct
Question 9 :  224.0.0.9 is a multicast address.
The answer is t
```







Qustion 4 :

```
class BankAccount:
    def __init__(self, account_number, account_holder, balance=0.0):
        self.account_number = account_number
```

```
        self.account_holder = account_holder
        self.balance = balance

    def deposit(self, amount):
        self.balance += amount
        print(f"Deposited ${amount}. Current balance:
${self.balance}")

    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
            print(f"Withdrew ${amount}. Current balance:
${self.balance}")
        else:
            print("Insufficient funds")

    def get_balance(self):
        return self.balance
```

```
class SavingsAccount(BankAccount):
    def __init__(self, account_number, account_holder, balance=0.0, interest_rate=0.0):
        super().__init__(account_number, account_holder, balance)
        self.interest_rate = interest_rate

    def apply_interest(self):
        interest_amount = self.balance * (self.interest_rate / 100)
        self.balance += interest_amount

    def print(self):
        print(f"Current balance: ${self.balance}, Interest rate: {self.interest_rate}%")


# Create an instance of BankAccount
bank_account = BankAccount("123456789", "John Doe")
```

```
bank_account.deposit(1000)
bank_account.withdraw(500)
print("Final balance:", bank_account.get_balance())


# Create an instance of SavingsAccount
savings_account = SavingsAccount("987654321", "Jane Smith", interest_rate=2.5)
savings_account.deposit(5000)
savings_account.apply_interest()
savings_account.print()
```

```
In [2]: runfile('C:/Users/Mahmoud Sakr/Desktop/واجب خلدونية/Homework/untitled0.py', wdir='C:/
Users/Mahmoud Sakr/Desktop/واجب خلدونية/Homework')
Deposited $1000. Current balance: $1000.0
Withdrew $500. Current balance: $500.0
Final balance: 500.0
Deposited $5000. Current balance: $5000.0
Current balance: $5125.0, Interest rate: 2.5%
```