

**Syrian Arab Republic**

**Lattakia – Tishreen  
University**

**Department of  
Communication and  
electrical engineering**

**5<sup>th</sup> , Network Programming  
: Homework No1**



الجمهورية العربية السورية

وزارة التعليم العالي والبحث العلمي

جامعة تشرين

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الاتصالات والالكترونيات

وظيفة مقرر برمجة الشبكات الأولى

إعداد الطالبة: ضحى خالد زنجري

الرقم الجامعي:

2402

إشراف الدكتور المهندس:

مهند عيسى

## 1. شرح السؤال الأول:

```
A.py > ...
1 L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']
2 L2 = [80, 443, 21, 53]
3 d = {x:y for x,y in zip(L1, L2)}
4 print(d)
```

```
OUTPUT TERMINAL PROBLEMS 2 PORTS DEVDB
PS C:\Users\ASUS\Desktop\ابرح >دمج & C:/Users/ASUS/
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
PS C:\Users\ASUS\Desktop\ابرح >دمج
```

تم تعريف القائمتين L1 و L2

عناصرها هي البروتوكولات والمنافذ

التي تنفذ منها (تعمل عليها)

باستخدام طريقة بناء القواميس

Dictionary Comprehension

أنشئت القاموس d لتكون المفاتيح فيه

هي البروتوكولات والقيم المقابلة هي أرقام المنافذ، تم ذلك باستخدام التابع zip(L1, L2) عن طريق القائمتين له وأخذ النتيجة المقابلة كل مرة بالاستفادة من الحلقة for.

## 2. شرح السؤال الثاني:

```
B.py > ...
1 num = eval(input('Please Enter Your Number to calc the factorial: '))
2 f = 1
3 if num < 0:
4     print("Your number is negative!")
5 if num == 0 or num == 1:
6     print(1)
7 while num > 0:
8     f *= num
9     num -= 1
10 print(f)
11
```

```
OUTPUT TERMINAL PROBLEMS 2 PORTS DEVDB D
PS C:\Users\ASUS\Desktop\ابرح >دمج & C:/Users/ASUS/A

Please Enter Your Number to calc the factorial: 5
120
PS C:\Users\ASUS\Desktop\ابرح >دمج
```

في هذا السؤال قمت بتخزين العدد المراد حساب العامل له في المتحول num، تم اختبار إن كانت قيمة العدد سالبة لنظهر ذلك، وإن كان العدد يساوي صفر أو واحد ف العامل له يساوي 1. باستخدام الحلقة while تم حساب العامل باستخدام المتحول محايد الضرب f. وإنقاص المتحول بمقدار 1 في كل دوران للحلقة.

3. شرح السؤال الثالث:

```
C.py > ...
1 L = ['Network', 'Bio', 'Programming', 'Physics', 'Music']
2 for i in L:
3     if i.startswith('B'):
4         print(i)
```

```
OUTPUT TERMINAL PROBLEMS 2 PORTS
PS C:\Users\ASUS\Desktop\ابر >دمج & C:
Bio
PS C:\Users\ASUS\Desktop\ابر >دمج
```

تم تعريف القائمة L وإضافة العناصر لها ثم

باستخدام الحلقة for والميثود startswith() اختبار الكلمات التي تبدأ ب B.

4. شرح السؤال الرابع:

```
D.py > ...
1 d = {x: x + 1 for x in range(0, 11)}
2 print(d)
```

```
OUTPUT TERMINAL PROBLEMS 2 PORTS DEVDB DEBUG CONSOLE
PS C:\Users\ASUS\Desktop\ابر >دمج & C:/Users/ASUS/AppData/Local/Prog
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
PS C:\Users\ASUS\Desktop\ابر >دمج
```

باستخدام الطريقة Dictionary Comprehension تم تعريف وبناء القاموس d المفاتيح هي المتغير x والقيم المقابلة  $x + 1$ .

5. شرح السؤال الخامس:

```
ges_2.py > ...
1  binary_Number = input("Please Write a binary number: ")
2
3  if not all(char in '01' for char in binary_Number):
4      raise("Invalid input, please enter a binary number")
5
6  else:
7      decimal = int(binary_Number, 2)
8      print("your decimal number: ", decimal)
```

```
OUTPUT  TERMINAL  PROBLEMS  2  PORTS  DE
PS C:\Users\ASUS\Desktop\ابرح >دمج & C:/Users
.py"
Please Write a binary number: 11011101010101
your decimal number: 14165
PS C:\Users\ASUS\Desktop\ابرح >دمج
```

تم أخذ العدد الثنائي من المستخدم

على شكل سلسلة محارفية، ثم

أجرينا اختبار إن كانت عناصر هذه السلسلة هي أرقام من النظام الثنائي (0,1) أم لا، إن لم تكن سنظهر رسالة خطأ. باستخدام الباني int() وبتمرير العدد المدخل ثم أساس النظام الثنائي 2 قمنا بتحويل العدد الثنائي لـ عشري.

## 6. شرح السؤال السادس:

```
qes_3.py > ...
1  import csv
2
3  try:
4      with open('my_qes.csv') as file:
5          reader = csv.reader(file)
6          questions = []
7          answers = []
8          for row in reader:
9              questions.append(row[0])
10             answers.append(row[1])
11 except FileNotFoundError:
12     print("The file 'my_qes.csv' was not found.")
13     exit()
14 except IndexError:
15     print("There was an error reading the 'my_qes.csv'. Please check the file format.")
16     exit()
17
18 name = input("Write Your name: ")
19
20 count = 0
21
22 for i in range(len(questions)):
23     answer = input(questions[i] + " ")
24     if answer.lower() == answers[i].lower():
25         print("correct")
```

```
my_qes.csv > data
1  what is your name,Ruba
2  How old are you,25
3  1 + 1,2
4  2 + 2,4
5  3 +3,6
6  4 + 4,8
7  5+5,10
8  6+6,12
9  7+7,14
```

تم وضع الأسئلة في ملف

my\_qes.csv حيث أن

العنصر ذو الفهرس 0 في

السطر يمثل السؤال

والعنصر ذو الفهرس 1 يمثل

الجواب. يتم تخزين اسم المستخدم الذي يجيب على الأسئلة مع نتيجته في الملف my\_ans.csv.

يتم التعامل من خلال الـ csv module في بايثون في برنامجي مع هذه البيانات وتخزينها في قائمة الأسئلة بعد جلبها. ونستخدم قائمة أخرى هي قائمة الأجوبة لتخزين النتيجة.

## 7. شرح السؤال السابع:

```
qes_4.py > ...
1  class BankAccount:
2      def __init__(self, account_number, account_holder, balance=0.0):
3          self.account_number = account_number
4          self.account_holder = account_holder
5          self.balance = balance
6
7      def deposit(self, amount):
8          self.balance += amount
9          print("Deposited ${}. New balance: {}".format(amount, self.balance))
10
11     def withdraw(self, amount):
12         if amount > self.balance:
13             print("Insufficient funds.")
14         else:
15             self.balance -= amount
16             print("Withdrew ${}. New balance: {}".format(amount, self.balance))
17
18     def get_balance(self):
19         return self.balance
20
21     def __str__(self):
22         return "Account Number: {}, Account Holder: {}, Balance: {}".format(
23             self.account_number, self.account_holder, self.balance
24         )
25
```

```

27 class SavingsAccount(BankAccount):
28     def __init__(self, account_number, account_holder, interest_rate, balance=0.0):
29         super().__init__(account_number, account_holder, balance)
30         self.interest_rate = interest_rate
31
32     def apply_interest(self):
33         interest = self.balance * self.interest_rate
34         self.deposit(interest)
35
36     def __str__(self):
37         return "{}, Interest Rate: {}".format(
38             super().__str__(), self.interest_rate
39         )
40
41
42 bank_account = BankAccount("1000000", "Doha_1")
43
44 bank_account.deposit(1000)
45
46 bank_account.withdraw(500)
47
48 print(bank_account)
49
50 savings_account = SavingsAccount("78000", "Doha_2", interest_rate=0.05)
51
52 savings_account.apply_interest()
53 print(savings_account)
54

```