



الجمهورية العربية السورية  
وزارة التعليم العالي والبحث العلمي  
جامعة تشرين  
كلية الهندسة الميكانيكية والكهربائية  
قسم هندسة الاتصالات والالكترونيات  
السنة الخامسة

# وظيفة برمجة الشبكات الأولى

هديل غياث إسماعيل 2827  
السنة الخامسة  
اتصالات

# First Network Programming Homework

## Question 1: Python Basics?

A- If you have two lists, L1=['HTTP','HTTPS','FTP','DNS'] L2=[80,443,21,53], convert it to generate this dictionary  
d={'HTTP':80,'HTTPS':443,'FTP':21,'DNS':53 }.

```
Qes1_A.py > ...
1  L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']
2  L2 = [80, 443, 21, 53]
3
4  d = {x: y for x, y in zip(L1, L2)}
5  print(d)
6
```

```
1/Qes1_A.py
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
```

B- Write a Python program that calculates the factorial of a given number entered by user.

```
Qes1_B.py > ...
1  number = eval(input("Enter a non-negative number to calculate the factorial: "))
2
3  fact = 1
4
5  if number < 0:
6      print("Sorry, factorial does not exist for negative numbers.")
7  else:
8      for i in range(1, number + 1):
9          fact *= i
10
11  print("The factorial of", number, "is", fact)
12
```

```
l/Qes1_B.py
Enter a non-negative number to calculate the factorial: 9
The factorial of 9 is 362880
```

C- L=['Network', 'Bio', 'Programming', 'Physics', 'Music'] In this exercise, you will implement a Python program that reads the items of the previous list and identifies the items that starts with 'B' letter, then print it on screen. Tips: using loop, 'len ()', startswith() methods.

```
Qes1_C.py > ...
1  L = ['Network', 'Bio', 'Programming', 'Physics', 'Music']
2
3  for item in L:
4      if item.startswith('B'):
5          print(item)
6
```

```
l/Qes1_C.py
Bio
```

D: Using Dictionary comprehension, Generate this dictionary  
d={0:1,1:2,2:3,3:4,4:5,5:6,6:7,7:8,8:9,9:10,10:11}

```
Qes1_D.py > ...
1  d = {key: key+1 for key in range(0, 11, 1)}
2
3  print(d)
```

```
l/Qes1_D.py
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
```

## شرح الإجابات:

1. تم تعريف القائمتين L1 و L2 ثم باستخدام طريقة بناء القواميس تم بناء قاموس d بحيث تم مقابلة أسماء البروتوكولات مع أسماء المنافذ باستخدام التابع zip() ثم باستخدام الحلقة for تم الدوران على كل ثنائية ووضع قيم ال key وال value.
2. يتم طلب من المستخدم إدخال عدد غير سالب باستخدام دالة input() ويتم تحويل القيمة المدخلة إلى عدد صحيح باستخدام int() وتخزينها في المتغير number. يتم تعيين قيمة ابتدائية للمتغير fact  $1 \leq$ . يتم التحقق مما إذا كان العدد الذي أدخله المستخدم سالباً أم لا. إذا كان العدد سالباً، يتم طباعة رسالة تفيد بأن عامل التسلسل غير موجود للأرقام السالبة. في حالة كان العدد غير سالب، يتم دخول الحلقة التكرارية for التي تقوم بضرب القيم من 1 إلى العدد المدخل بواسطة المستخدم. تُحسب قيمة عامل التسلسل باستخدام العملية  $fact *= i$ ، حيث تُحدث قيمة fact في كل دورة بضربها بقيمة i.
3. تم تعريف القائمة L وإضافة العناصر لها ثم باستخدام الحلقة for والميثود startswith() اختبار الكلمات التي تبدأ ب B.
4. باستخدام الطريقة Dictionary Comprehension تم تعريف وبناء القاموس d المفاتيح هي المتغير x والقيم المقابلة  $x + 1$ .

## Question 2: Convert from Binary to Decimal

```
Qes2.py > main
1 def binary_to_decimal(binary):
2
3     if not binary.isdigit():
4         raise ValueError("Invalid binary number.")
5
6     decimal = 0
7     for digit in binary:
8         decimal = decimal * 2 + int(digit)
9
10    return decimal
11
12
13 def main():
14     binary = input("Enter a binary number: ")
15
16     decimal = binary_to_decimal(binary)
17
18     print("The decimal equivalent of", binary, "is", decimal)
19
20 if __name__ == "__main__":
21     main()
```

```
1/Qes2.py
Enter a binary number: 01000011110
The decimal equivalent of 01000011110 is 542
```

تعريف دالة: `binary_to_decimal(binary)`

- يتم تمرير النص الثنائي كمدخل للدالة.
- يتم التحقق إذا كان النص الثنائي يتكون من أرقام ثنائية بواسطة الدالة `isdigit()`، إذا لم يكن كذلك فإنه يتم رفع خطأ برمجي `ValueError` برسالة "Invalid binary number."
- يتم تهيئة المتغير `decimal` بقيمة ابتدائية 0.

- يتم تحويل كل رقم ثنائي في النص الثنائي إلى عدد عشري من خلال الشيفرة  

$$\text{decimal} = \text{decimal} * 2 + \text{int}(\text{digit}).$$
- يتم إرجاع القيمة العشرية المحسوبة.  
 تعريف الدالة: `main()`
- يطلب من المستخدم إدخال رقم ثنائي من خلال `input()` ويخزن في المتغير `binary`.
- يتم استدعاء دالة `binary_to_decimal(binary)` لتحويل الرقم الثنائي إلى رقم عشري وتخزين القيمة في المتغير `decimal`
- يتم طباعة القيمة العشرية المحسوبة مع النص الثنائي الأصلي.  
 تحديد نقطة البداية للتنفيذ:
- تحقق من أن نقطة الدخول `__name__` تكون `"__main__"` لتشغيل الدالة الرئيسية `main()`

### Question 3: Working with Files" Quiz Program"

```
Qes3.py > ...
1  import csv
2
3  def load_questions_from_csv(file_path):
4      questions = []
5      with open(file_path, 'r') as file:
6          reader = csv.reader(file)
7          for row in reader:
8              questions.append(row)
9      return questions
10
11 def quiz_user(questions):
12     score = 0
13     for q in questions:
14         print(q[0])
15         user_answer = input("Your answer: ")
16         if user_answer.lower() == q[1].lower():
17             score += 1
18     return score
19
20 def save_result_to_csv(user_name, score):
21     with open('quiz_results.csv', 'a') as file:
22         writer = csv.writer(file)
23         writer.writerow([user_name, score])
24
25 file_path = 'quiz_questions.csv'
26 questions = load_questions_from_csv(file_path)
27
28 user_name = input("Enter your name: ")
29 user_score = quiz_user(questions)
30
31 print(f"Quiz completed! Your score: {user_score}/{len(questions)}")
32 save_result_to_csv(user_name, user_score)
```

```

quiz_questions.csv > data
1  What is the value of pi to two decimal places?,3.14
2  What is the square root of 81?,9
3  What is 5 multiplied by 8?,40
4  What is the sum of 12 and 25?,37
5  What is 3 squared?,9
6  What is the next prime number after 17?,19
7  What is the value of log(1)?,0
8  What is the product of 7 and 9?,63
9  What is 20 divided by 4?,5
10 What is the cube of 3?,27
11 Who painted the Mona Lisa?,Leonardo da Vinci
12 What is the largest mammal in the world?,Blue whale
13 What is the longest river in the world?,Nile River
14 Which country is known for the Great Barrier Reef?,Australia
15 What is the main ingredient in guacamole?,Avocado
16 Who is known as the Father of Computers?,Charles Babbage
17 What is the capital of Italy?,Rome
18 Which bird is a symbol of peace?,Dove
19 What is the hottest planet in our solar system?,Venus
20 What is the main ingredient in sushi rolls?,Rice

```

تم استيراد المودل CSV للتعامل مع ملفات الأسئلة والأجوبة من النوع CSV، باستخدام التابع `load_questions_from_csv()` تم تحميل الأسئلة من الملف `quiz_questions.csv` وتخزينها في القائمة `questions` للتعامل معها. بعد قراءة الملف ومعالجة الأسئلة يتم استقبال إجابات المستخدم عن طريق التابع `quiz_user()`، وزيادة المتحول `score` في كل مرة يجيب المستخدم إجابة صحيحة.



#### Question 4: Object-Oriented Programming - Bank Class

Define a class BankAccount with the following attributes and methods:  
Attributes: account\_number (string), account\_holder (string), balance (float, initialized to 0.0) Methods: deposit(amount), withdraw(amount) , get\_balance() - Create an instance of BankAccount, - Perform a deposit of \$1000, - Perform a withdrawal of \$500. - Print the current balance after each operation. - Define a subclass SavingsAccount that inherits from BankAccount and adds interest\_rate Attribute and apply\_interest() method that Applies interest to the balance based on the interest rate. And Override print() method to print the current balance and rate. - Create an instance of SavingsAccount , and call apply\_interest() and print() functions.

Qes4.py > SavingsAccount

```
1 class BankAccount:
2     def __init__(self, account_number, account_holder):
3         self.account_number = account_number
4         self.account_holder = account_holder
5         self.balance = 0.0
6
7     def deposit(self, amount):
8         self.balance += amount
9
10    def withdraw(self, amount):
11        if amount <= self.balance:
12            self.balance -= amount
13        else:
14            print("Insufficient funds")
15
16    def get_balance(self):
17        return self.balance
18
19 account1 = BankAccount("12345", "Hadeel_1")
20
21 account1.deposit(1000)
22 print("Balance after deposit: $", account1.get_balance())
23
24 account1.withdraw(500)
25 print("Balance after withdrawal: $", account1.get_balance())
```

الخرج:

```
Balance after deposit: $ 1000.0
Balance after withdrawal: $ 500.0
Current balance: $ 2100.0
Interest rate: 0.05
```

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

    def apply_interest(self):
        interest_amount = self.balance * self.interest_rate
        self.deposit(interest_amount)

    def print_info(self):
        print("Current balance: $", self.get_balance())
        print("Interest rate: ", self.interest_rate)

savings_account1 = SavingsAccount("54321", "Hadeel_2", 0.05)
savings_account1.deposit(2000)
savings_account1.apply_interest()
savings_account1.print_info()
```