

Syrian Arab Republic

Lattakia - Tishreen University

Department of Communication and electrical  
engineering

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



الجمهورية العربية السورية

اللاذقية - جامعة تشرين

كلية الهندسة الكهربائية والميكانيكية

قسم هندسة الاتصالات والالكترونيات

السنة الخامسة: وظيفة 1 برمجة شبكات

## بتول علي حسن 2885

### Q1:A-

```
L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']
L2 = [80, 443, 21, 53]
d={}
for i in range(len(L1)):
    d[L1[i]]=L2[i]
print(d)
```

الخرج

```
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
```

Process finished with exit code 0

### Q1:B-

```
while True:
    s = int(input("Enter a number "))
    if s < 0:
        break
    fa=1
    for n in range(1,s+1):
        fa=fa*n
    print(s,": ",fa)
```

الخرج

```
Enter a number 8
8 : 40320
Enter a number 2
2 : 2
Enter a number 5
5 : 120
Enter a number 3
3 : 6
Enter a number -88
```

Process finished with exit code 0

### Q1:C-

```
L = ['Network', 'Bio', 'Programming', 'Physics', 'Music']
for x in L:
    if x.startswith('B'):
        print(x)
```

الخرج

Bio

Process finished with exit code 0

Q1:D-

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

الخرج

```
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
```

Process finished with exit code 0

Q2:

```
while True:
    b = input("Enter binary to convert to decimal and 0 to exit: ")
    if b == '0':
        print("stopped")
        break
    if b.isdigit():
        if ('1' not in b) and ("0" not in b):
            print("error")
            continue
        else:
            l = []
            dec = 0
            for i in b:
                l.append(int(i))
            l.reverse()
            for i in range(len(l)):
                dec += l[i] * 2 ** i
            print(dec)
    else:
        print("error input")
```

الخرج

```
Enter binary to convert to decimal and 0 to exit: 1111111
127
Enter binary to convert to decimal and 0 to exit: 111111111111
4095
Enter binary to convert to decimal and 0 to exit: 111
7
Enter binary to convert to decimal and 0 to exit: 1101
13
Enter binary to convert to decimal and 0 to exit: 0
stopped
```

Process finished with exit code 0

Q3:

```
import json
user_name = input("Enter your name: ")
infile = 'questions.txt'
outfile = 'result.json'
questions = []
answers=[]
with open(infile, 'r') as s:
    lines = s.readlines()
    for i in range(0, len(lines), 2):
        question = lines[i].strip()
        answer = lines[i+1].strip()
        questions.append(question)
        answers.append(answer)
mark = 0
for i in questions:
    s = input(i)
    if s.lower() == answers[questions.index(i)].lower():
        mark += 1
d={"name":user_name,"mark":str(mark)}
with open(outfile, 'w') as f:
    json.dump(d, f)

print(f"{user_name}, your score is: {mark}/{len(questions)}")
```

الكود يبدأ بطلب اسم المستخدم، ثم يقرأ الأسئلة والإجابات من ملف "questions.txt" ويخزنها في قائمتي "questions" و "answers". ثم يبدأ الكود في طرح الأسئلة على المستخدم ويقوم بمقارنة إجابات المستخدم مع الإجابات الصحيحة المخزنة في قائمة "answers". يتم تسجيل الإجابات الصحيحة في متغير "mark".

يتم إنشاء المتغير d الذي يحتوي على اسم المستخدم ونتيجته، والذي يتم تخزينه في ملف "result.json" باستخدام مكتبة "json".

#### questions.txt:

```
1-A.py × 1-B.py × 1-C.py × 1-D.py × 2.py × 3.py × questions.txt ×
1 What is the capital of Syria?
2 damascus
3 What is the currency of syria?
4 lera
5 What is the tallest mountain in the world?
6 Everest
7 What is 5 + 2?
8 7
9 What is 5 + 3?
10 8
11 What is the square root of 25?
12 5
13 What is 7 * 7?
14 49
15 What is the capital of Syria?
16 damascus
17 What is the currency of syria?
18 lera
19 What is the tallest mountain in the world?
20 Everest
21 What is 5 + 2?
22 7
23 What is 5 + 3?
24 8
25 What is the square root of 25?
26 5
27 What is 7 * 7?
28 49
```

#### mark.json

1.py × 1-B.py × 1-C.py × 1-D.py × 2.py × 3.py × questions.txt × result.json ×

```
{ "name": "Batoul", "mark": "12" }
```

Q4:

# فائدة بدون البنك حساب كلاس تعريف

```
class BankAccount:
    def __init__(self, account_number, account_holder):
        # البنك الحساب رقم
        self.account_number = account_number
        # الحساب صاحب اسم
        self.account_holder = account_holder
        # الوظيفة نص حسب الافتراضي الرصيد
        self.balance = 0.0

    # المودع المبلغ بقيمة الرصيد زيادة تابع
    def deposit(self, amount):
        self.balance += amount
        # الرصيد من المسحوب المبلغ خصم تابع
    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
        else:
            print("لاحقاً المحاولة يرجى خطأ حدث")

    # الرصيد لمعرفة تابع
    def get_balance(self):
        return self.balance
```

# فائدة مع البنك حساب كلاس تعريف

```
# BankAccount الكلاس يرث الكلاس
class SavingsAccount(BankAccount):
    # الاب الابني التابع الى الانتقال
    def __init__(self, account_number, account_holder, interest_rate):
        super().__init__(account_number, account_holder)
        self.interest_rate = interest_rate # الفائدة نسبة

    # الفائدة نسبة على بناء كفاءة المستحق المبلغ لحساب تابع
    def apply_interest(self):
        interest = self.balance * self.interest_rate / 100
        self.balance += interest # كفاءة المستحق المبلغ بقيمة الرصيد زيادة

# override لتابع print
    def __str__(self):
        p="الحالي الرصيد:" +str(self.balance)+"الفائدة نسبة:"
        "+str(self.interest_rate)
        return p
```

# غرض اشتقاق

```
bank = BankAccount("2971", "شنانى لين")
```

```
# دولار 1000 إيداع
bank.deposit(1000)
print("الرصيد الحالي:", bank.get_balance())
# دولار 500 سحب
bank.withdraw(500)
print("الرصيد الحالي:", bank.get_balance())
savings= SavingsAccount("2222", "محمد احمد", 5)
# دولار 10000 إيداع
savings.deposit(10000)
print(savings)
# الفائدة تطبيق
savings.apply_interest()
print(savings)
```

الخرج

```
الرصيد الحالي: 1000.0
الرصيد الحالي: 500.0
الرصيد الحالي: 10000.0نسبة الفائدة: 5
الرصيد الحالي: 10500.0نسبة الفائدة: 5
```

Process finished with exit code 0