

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

الرقم الجامعي: 2461

الاسم: ديانا عدنان سلوم

### السؤال الأول

-A-

```
L1=['HTTP','HTTPS','FTP','DNS']
```

```
L2=[80,443,20,53]
```

```
d={}
```

```
for i in range(len(L1)):
```

```
    d[L1[i]]=L2[i]
```

```
print(d)
```

الخرج

```
{ 'HTTP': 80, 'HTTPS': 443, 'FTP': 20, 'DNS': 53 }
```

-B-

```
def f(x):
```

```
    fa=1
```

```
    for i in range(1,x+1):
```

```
        fa*=i
```

```
        if i==x:
```

```
            break
```

```
    return fa
```

```
while True:
```

```
    num=int(input("enter positive number to calculate it factorial: "))
```

```
    if num >=0:
```

```
        print(f(num))
```

```
    else:
```

```
        print("enter positive number")
```

الخرج

```
enter positive number to calculate it factorial: 5
120
enter positive number to calculate it factorial: 6
720
enter positive number to calculate it factorial: 0
1
enter positive number to calculate it factorial: -1
enter positive number
```

---

-C-

```
#C
L=['Network','Bio','Programming','Physics','Music']
for i in range(len(L)):
    if L[i].startswith('B'):
        print("the item which sarts with B is:",L[i])
```

الخرج

the item which sarts with B is: Bio

---

-D-

```
d={x:x+1 for x in range(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}

---

### السؤال الثاني

**while True:**

```
bin= input("input an binary number and enter e to exit: ")
```

```
if bin=="e":
```

```
    break
```

```
t=0
```

```
for i in bin:
```

```
    if i == '0' or i == '1':
```

```
        pass
```

```
    else:
```

```
        t=1
```

```
        print('error')
```

```
        break
```

```
if t==0:
```

```
    dec = 0
```

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

```
        digit = int(bin[i])
```

```
        dec += digit * 2**(len(bin)-i-1)
```

```
    print(dec)
```

يطلب الكود من المستخدم إدخال رقم ثنائي، ثم يقوم بتحويله إلى رقم عشري بالشكل التالي:

1. البرنامج يدخل في حلقة `while True` التي تستمر حتى يقوم المستخدم بإدخال "e" للخروج.
2. في كل مرة، البرنامج يطلب من المستخدم إدخال رقم ثنائي ويخزنه في متغير `bin`.
3. البرنامج يقوم بالتحقق من أن الرقم المدخل يتكون فقط من الأرقام 0 و 1. إذا كان هناك أي رمز آخر، يتم طباعة "error" والخروج من الحلقة.
4. إذا كان الرقم المدخل صالحًا، البرنامج يقوم بتحويله إلى رقم عشري. وذلك عن طريق:
  - تهيئة متغير `dec` إلى 0.
  - التكرار على أرقام الرقم الثنائي، بدءًا من الأيسر.
  - لكل رقم، يتم ضربه في 2 مرفوعًا إلى القوة المناسبة، ثم إضافة الناتج إلى `dec`.
  - طباعة القيمة النهائية لـ `dec` كرقم عشري.
5. ثم تعود الحلقة إلى البداية و ينتظر المستخدم إدخال رقم ثنائي آخر.

### الخرج

```
input an binary number and enter e to exit: 5
error
input an binary number and enter e to exit: 1111
15
input an binary number and enter e to exit: 0
0
input an binary number and enter e to exit: d
error
input an binary number and enter e to exit: 1101
13
input an binary number and enter e to exit: 001
1
input an binary number and enter e to exit: e
```

### السؤال الثالث

نعرف ملف الأسئلة بالشكل:

split method to turn string in list\$true

join method to turn list in string\$true

34\$2+32

34\$1+33

what is my university id\$2461

34\$4+30

..

.

وهو ملف نصي اسمه q.txt

```
print("answer a questions")
name=input("enter your name")
infile=open("q.txt",'r')
outfile=open("Diana.csv",'w')
s=infile.read()
```

```

l=s.splitlines()
infile.close()
n=0
for i in l:
    t=i.split('$')
    print(t[0])
    s=input()
    if s==t[1]:
        n+=1
print("welcome",name,"your result is", n," OF",len(l))
outfile.write(name)
outfile.write(",")
outfile.write(str(n))
outfile.write(",")
outfile.write(" OF "+str(len(l)))
outfile.close()

```

1. يطبع الرسالة "answer a questions" لإبلاغ المستخدم بأنه سيتم طرح أسئلة عليه.

2. يطلب من المستخدم إدخال اسمه ويخزنه في المتغير name.

3. يفتح ملف "q.txt" بوضع القراءة ويأخذ محتواه في المتغير s.

4. ينقسم محتوى ملف "q.txt" إلى قائمة من الأسطر (كل سؤال مع إجابته) ويخزنها في المتغير l.

5. يفتح ملف "Diana.csv" بوضع الكتابة للحفاظ على نتيجة المستخدم.

6. يبدأ في تكرار الأسئلة من القائمة l:

- يطبع السؤال (الجزء قبل الرمز \$)
- يطلب من المستخدم الإجابة ويخزنها في المتغير s
- يقارن إجابة المستخدم مع الإجابة الصحيحة (الجزء بعد الرمز \$)
- إذا كانت الإجابة صحيحة، يزيد قيمة المتغير n بواحد

7. بعد الانتهاء من الأسئلة، يطبع رسالة للمستخدم تعلمه بنتيجته.

8. يكتب اسم المستخدم وعدد الإجابات الصحيحة والإجمالي في ملف "Diana.csv".

9. يغلق ملف "Diana.csv".

الخرج

```
answer a questions
enter your nameDiana
split method to turn string in list
true
join method to turn list in string
true
32+2
34
33+1
34
what is my university id
2461
30+4
34
35-1
34
2+1000
1002
what is my university id
2461
16+1
17
8+2
20
2*2
4
15*10
150
0+0
0
12+13
22
welcome Diana your result is 13 OF 15
```

## السؤال الرابع

نعرف كلاسين بايثون `BankAccount` و `SavingsAccount` الكلاس `BankAccount` يمثل حساب بنكي عادي، والذي له رقم الحساب، اسم صاحب الحساب، وكذلك الرصيد الحالي. هذا الكلاس له ثلاث وظائف `deposit()`: لإضافة مبلغ إلى الرصيد، `withdraw()` لخصم مبلغ من الرصيد، و `get_balance()` للحصول على رصيد الحساب الحالي.

الكلاس `SavingsAccount` هو كلاس فرعي من الكلاس `BankAccount`، وهو يمثل حساب توفير. بالإضافة إلى المعلومات الموجودة في كلاس `BankAccount`، فإن `SavingsAccount` له أيضًا معدل الفائدة من خلال وظيفة `apply_interest()` التي تُضيف الفائدة إلى الرصيد، و `override` للتابع `__str__()` من أجل طباعة الرصيد الحالي والفائدة.

في نهاية الكود، يتم إنشاء كائنين: واحد من كلاس `BankAccount` باسم "Diana" والآخر من كلاس `SavingsAccount` باسم "Diana1" يتم إجراء عمليتي إيداع وسحب على الحساب العادي، ثم يتم استدعاء وظيفة `apply_interest()` على حساب التوفير وطباعة قيمة الرصيد والفائدة.

بالنسبة ل `override` للتابع `__str__()` في كلاس `SavingsAccount` هذا التابع يُعرّف كيفية طباعة الكائن من هذه الكلاس عندما يُطبع مباشرة. في هذه الحالة، يتم طباعة الرصيد الحالي والفائدة بطريقة مخصصة.

```
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):
        if amount>0:
            self._balance+=amount
            print("current balance after deposit is: ", self.get_balance())
    def withdraw(self,amount):
        if amount>0:
            if self._balance>=amount:
                self._balance-=amount
                print("current balance after withdraw is: ",self.get_balance())
            else:
                print("No Enough money in account")
    def get_balance(self):
        return self._balance

class SavingsAccount(BankAccount):
    def __init__(self,account_number,account_holder,balance=0,interest_rate=0):
        super().__init__(account_number,account_holder,balance)
        self._interest_rate=interest_rate
    def apply_interest(self):
        self._balance+=self._balance*self._interest_rate
```

```
#Override print() method to print the current balance and rate.
def __str__(self):
    return f"Current balance is: "+ str(self._balance)+ " interest rate is: "+ str(self._interest_rate)

#Create an instance of BankAccount, - Perform a deposit of $1000, - Perform a withdrawal of $500.
Diana=BankAccount("2461", "Diana Saloum")
Diana.deposit(1000)
Diana.withdraw(500)

#Create an instance of SavingsAccount, and call apply_interest() and print() functions.
Diana1=SavingsAccount("0001", "Diana",1500,0.1)
Diana1.apply_interest()
print(Diana1)
```

الخرج

```
current balance after deposit is:  1000.0
current balance after withdraw is:  500.0
Current balance is: 1650.0 interest rate is: 0.1
```

---