

Syrian Arab Republic

Lattakia - Tishreen University

Department of Communication and electrical
engineering

5th , Network Programming : Homework No1



الجمهورية العربية السورية

الملاذقية - جامعة تشرين

كلية الهندسة الكهربائية والميكانيكية

قسم هندسة الاتصالات والإلكترونيات

السنة الخامسة: وظيفة 1 برمجة شبكات

Name: مايا محمد حسن , Number 2955

First Network Programming Homework

Question 1: Python Basics?

A:

```
def dd(k,v):
```

```
    d=dict()
```

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

```
        d[k[i]]=v[i]
```

```
    return d
```

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

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

```
print(dd(L1,L2))
```

output:

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

B:

نعرف تابع اسمه **func** يقوم بحساب العامل (الضرب التراكمي) لأي عدد معطى. العامل لعدد ما هي حاصل ضرب جميع الأعداد الموجبة من 1 إلى ذلك العدد.

إذا كان العدد المعطى للتابع هو 0، فإنه يعيد القيمة 1، لأن العامل لـ 0 محددة بـ 1.

أما إذا كان العدد أكبر من 0، فإن التابع يبدأ بتخزين القيمة 1 في متغير اسمه s، ثم يستخدم حلقة **for** لضرب جميع الأعداد من 1 إلى العدد المعطى، ويعيد القيمة النهائية.

بعد ذلك يدخل البرنامج في حلقة **while** حتى يدخل المستخدم رقم سالب. داخل هذه الحلقة يطلب البرنامج من المستخدم إدخال رقم، ويستدعي التابع **func** ويطبع النتيجة على الشاشة.

إذا أدخل المستخدم قيمة غير صحيحة (مثل قيمة غير رقمية)، فإن كتلة **except** تعالج هذا الخطأ وتطبع " **Error** **Input** " على الشاشة.

```
def func(x):
    if x == 0:
        return 1
    else:
        s=1
        for i in range(1,x+1):
            s*=i
        return s
while True:
    try:
        number = int(input("Enter a number "))
        if number < 0:
            break
        result = func(number)
        print(f"The factorial of {number} is {result}")
    except:
        print("Error Input")
```

output:

```
Enter a number 0
The factorial of 0 is 1
Enter a number 1
The factorial of 1 is 1
Enter a number 2
The factorial of 2 is 2
Enter a number 5
The factorial of 5 is 120
Enter a number 10
The factorial of 10 is 3628800
Enter a number a
Error Input
Enter a number -5
```

C:

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

output:

```
['Bio']
```

D:

```
d = {i: i + 1 for i in range(11)}  
print(d)
```

output:

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

Question 2: Convert from Binary to Decimal

الكود يطلب من المستخدم إدخال رقم ثنائي وحرف "s" للتوقف. ثم يدخل في حلقة while حتى يقوم المستخدم بإدخال حرف "s".

داخل الحلقة، يتم تحويل الرقم الثنائي المدخل إلى رقم عشري كالتالي:

- يتم تعيين متغير decimal بالقيمة 0 في البداية.
- يتم استخدام حلقة for لتكرار كل رقم في الرقم الثنائي المدخل.
- إذا كان الرقم هو "1"، يتم إضافة قوة 2 بقوة تساوي طول الرقم الثنائي ناقص الرقم الحالي في الحلقة ناقص 1 إلى قيمة decimal.
- بعد الانتهاء من الحلقة يتم طباعة الرقم العشري المحسوب.
- إذا قام المستخدم بإدخال قيمة غير صحيحة (أي ليس رقم ثنائي)، يتم التقاط استثناء ValueError وطباعة "Invalid binary number".

بعد ذلك يطلب البرنامج من المستخدم إدخال رقم ثنائي آخر أو حرف "s" للتوقف.

وينتهي "End"، يطبع البرنامج "s" عند إدخال حرف

```
a = input("Enter a binary number and s to stop: ")
```

```
while a != "s":
```

```
    try:
```

```
        decimal = 0
```

```
        for i, digit in enumerate(a):
```

```
            if digit == '1':
```

```
                decimal += 2 ** (len(a) - i - 1)
```

```
        print(f"The decimal of {a} is {decimal}")
```

```
    except ValueError:
```

```
        print("Invalid binary number")
```

```
    a = input("Enter a binary number and s to stop: ")
```

```
print("End")
```

output:

```
Enter a binary number and s to stop: 101
The decimal of 101 is 5
Enter a binary number and s to stop: 10101
The decimal of 10101 is 21
Enter a binary number and s to stop: s
End
```

Question 3: Working with Files” Quiz Program”

هذا البرنامج يقوم بإجراء اختبار بسيط للطلاب باستخدام ملف "questions.csv" الذي يحتوي على قائمة الأسئلة والإجابات. يطلب البرنامج من المستخدم إدخال اسم الطالب. ثم يفتح ملفي "questions.csv" و"mark.csv"، حيث يستخدم الملف الأول لاستخراج قوائم الأسئلة والإجابات، والملف الثاني لتسجيل درجة الطالب. بعد ذلك يبدأ البرنامج في عرض الأسئلة واحداً تلو الآخر، وطلب الإجابة من الطالب. إذا كانت إجابة الطالب صحيحة، يتم زيادة درجته بمقدار 1. بعد الانتهاء من جميع الأسئلة، يطبع البرنامج اسم الطالب والعلامة التي حصل عليها، ثم يكتب هذه المعلومات في ملف "mark.csv".

```
def extractfield(filename,n):
    infile=open(filename,"r")
    return [line.rstrip().split(',')[n-1] for line in infile]
mark=0
name=input("enter student name: ")
infile=open("questions.csv",'r')
outfile=open("mark.csv","w")
questions=extractfield("questions.csv",1)
answers=extractfield("questions.csv",2)
for i in range(len(questions)):
    print(questions[i])
    answer=input("the answer is: ")
    if answer==answers[i]:
        mark+=1
print("name: ",name,"mark: ", mark)
outfile.write(name+','+str(mark))
outfile.close()
```

output:

```
the answer is: 0
2*2
the answer is: 0
3*3
the answer is: 9
0 % 3
the answer is: 3
2 % 1
the answer is: 0
4*4
the answer is: 16
1*1
the answer is: 1
2*2
the answer is: 4
name: مايا محمد حسن mark: 19
```

File of questions:

1*1	1
2*2	4
3*3	9
4*4	16
1*1	1
2*2	4
3*3	9
4*4	16
1-Jan	0
1+1	2
5 % 3	2
0 % 3	3
2 % 1	0
2*2	4
3*3	9
0 % 3	3
2 % 1	0
4*4	16
1*1	1
2*2	4

File of mark:

A	B	C	D
مايا محمد حسن	19		

Question 4: Object-Oriented Programming - Bank Class

هذا البرنامج يعرف كلاسين: "BankAccount" و "SavingsAccount". الكلاس "BankAccount" له ثلاث متغيرات: "account_number" و "account_holder" و "balance". عند إنشاء كائن "BankAccount" جديد، يتم تحديد هذه المتغيرات الثلاثة. كما يتضمن الكلاس "BankAccount" ثلاث طرق: "deposit" لإضافة مبلغ إلى الرصيد، و "withdraw" لسحب مبلغ من الرصيد، و "get_balance" للحصول على الرصيد الحالي.

أما الكلاس "SavingsAccount" فهو كلاس فرعي من "BankAccount"، أي أنه يرث كل خصائص ووظائف الكلاس الأب. عند إنشاء كائن "SavingsAccount" جديد، يتم تحديد "account_number" و "account_holder" و "balance" بالإضافة إلى "interest_rate"، وهي نسبة الفائدة السنوية. الكلاس "SavingsAccount" يحتوي على طريقة "apply_interest" لإضافة الفائدة المستحقة إلى الرصيد. كما تم تعريف طريقة "str" لإرجاع تمثيل نصي للكائن "SavingsAccount" يُظهر الرصيد الحالي ونسبة الفائدة.

في النهاية، يتم إنشاء كائنين: "Maya" من نوع "BankAccount" و "Maya_s" من نوع "SavingsAccount"، وإجراء بعض العمليات عليهما كالإيداع والسحب وإضافة الفائدة.

```
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 amount <= self.balance:
            self.balance -= amount
            print(f"Withdrawn {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=0.01):
        super().__init__(account_number, account_holder, balance)
        self.interest_rate = interest_rate

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

    def __str__(self):
        return f"Current balance: {self.balance}, Interest rate: {self.interest_rate}"

Maya=BankAccount("2955","Maya")
Maya.deposit(1000)
Maya.withdraw(500)
print(Maya.get_balance())
Maya_s=SavingsAccount("178487","Maya",interest_rate=0.22)
Maya_s.deposit(400)
Maya_s.apply_interest()
print(Maya_s)
```

output:

```
Deposited 1000. Current balance: 1000.0
Withdrawn 500. Current balance: 500.0
500.0
Deposited 400. Current balance: 400.0
Current balance: 488.0, Interest rate: 0.22
```
