

**BRAC UNIVERSITY**

**Department of Computer Science and Engineering**

**Examination**: Final **Duration**: 90 Minutes **No. of Questions**: 3

CSE 111: Programming Language II**Semester**: Fall 2022

**Full Marks**: 30

**No. of Pages**: 3

| Name:  (Please write in CAPITAL LETTERS) | ID: | Section: |
| --- | --- | --- |

**A**

✔ Use the back **part** of the answer script for rough work. **No washroom breaks.** ✔ At the end of the exam, put the question **paper** inside the answer script and **return both**.

**Question 1: CO4 [10 Points]**

**Design** the **Netflix** class with necessary properties so that the given output is produced.

| **#Write your code here**  s1 = Netflix("Wednesday",["Mystery","Supernatural"],15) print("==========1==========")  print(s1)  s2 = Netflix("Dark",["Mind-Bending","Sci-fi"]) print("==========2==========")  print(s2)  print("==========3==========")  Netflix.printDetails()  s3 = Netflix("Suits",["Comedy","Courtroom"],20) print("==========4==========")  print(s3)  s4 = Netflix("Demon Slayer",["Anime"],22)  print("==========5==========")  print(s4)  print("==========6==========")  Netflix.printDetails() | **Output:**  ==========1==========  Show name: Wednesday  Episodes: 15  Genre: Mystery, Supernatural ==========2==========  Show name: Dark  Episodes: 10  Genre: Mind-Bending, Sci-fi ==========3==========  Total number of shows: 2  Wednesday  Dark  ==========4==========  Show name: Suits  Episodes: 20  Genre: Comedy, Courtroom  ==========5==========  Show name: Demon Slayer  Episodes: 22  Genre: Anime  ==========6==========  Total number of shows: 4  Wednesday  Dark  Suits  Demon Slayer |
| --- | --- |

**Page 1 of 3 Set-A**

**Question 2: CO5 [10 Points]**

**Implement** the required class with the necessary properties to produce the given output for the following driver code.

[**Hints:**

1. You can only make a call to numbers starting with any of the given country codes.

2. In order to make a call, the following steps must be followed sequentially:

[Check sim card status –> Check available balance –> Check country code.]

| class Mobile:  countryCodes = {"880": "Bangladesh", "966": "India", "455": "USA"}  def \_\_init\_\_(self, model, simCardStatus): self.model = model  self.\_\_simCardStatus = simCardStatus  print(f"Model {model} is manufactured.") def setSimCardStatus(self,status):  self.\_\_simCardStatus = status  print("SIM card status updated successfully.") def getSimCardStatus(self):  return self.\_\_simCardStatus  def \_\_str\_\_(self):  return f"Mobile Phone Detail:\nModel: {self.model}\nSIM Card Status: {self.\_\_simCardStatus}"  **#Write your code here**  N3110 = Nokia("N3110", False)  print("#######################################") print(N3110)  print("1======================================") N1100 = Nokia("N1100", True,100)  print("#######################################") print(N1100)  print("2======================================") print(N3110.dialCall("88017196xxxx"))  print("3======================================") N3110.changeSIMCardStatus()  print("4======================================") print(N3110.dialCall("88017196xxxx"))  print("5======================================") N3110.rechargeSIMCard(200)  print("6======================================") print(N3110.dialCall("88017196xxxx"))  print("7======================================") print(N1100.dialCall("45617196xxxx"))  print("8======================================") print(N1100.dialCall("45517196xxxx"))  print(N1100.dialCall("96617196xxxx"))  print("9======================================") print(f"Dial call history for {N1100.model}: {N1100.dialCallHistory}")  print(f"Dial call history for {N3110.model}: {N3110.dialCallHistory}") | **Output:**  Model N3110 is manufactured. ################################### Mobile Phone Detail:  Model: N3110  SIM Card Status: False  Balance:0 TK  1================================= Model N1100 is manufactured. ################################### Mobile Phone Detail:  Model: N1100  SIM Card Status: True  Balance:100 TK  2================================= No SIM card available!  3================================= SIM card status updated  successfully.  4================================= Insufficient balance!  5================================= Recharge successful! Current balance 200 TK.  6================================= Dialing the number 88017196xxxx to Bangladesh region.  7================================= Dialing is not allowed in this region.  8================================= Dialing the number 45517196xxxx to USA region.  Dialing the number 96617196xxxx to India region.  9================================= Dial call history for N1100: ['45517196xxxx', '96617196xxxx'] Dial call history for N3110: ['88017196xxxx'] |
| --- | --- |

**Page 2 of 3 Set-A**

**Question – 3: CO4 [10 Points]**

| **1** | **class A:** |
| --- | --- |
| **2** | **temp = 7** |
| **3** | **def \_\_init\_\_(self):** |
| **4** | **self.y = A.temp - 3** |
| **5** | **self.sum = self.temp + 2** |
| **6** | **A.temp += 3** |
| **7** | **def methodA(self, m, n, x=0):** |
| **8** | **self.y = self.y + m + (A.temp)** |
| **9** | **x = x + 2 + n** |
| **10** | **self.sum = self.sum + x + self.temp** |
| **11** | **print(x, self.y, self.sum)** |
| **12** | **class B(A):** |
| **13** | **temp = 1** |
| **14** | **def \_\_init\_\_(self, obj=None):** |
| **15** | **super().\_\_init\_\_()** |
| **16** | **self.temp = self.temp + B.temp** |
| **17** | **self.sum = 3 + B.temp + A.temp** |
| **18** | **if obj != None:** |
| **19** | **obj.methodB(3, 6)** |
| **20** | **else:** |
| **21** | **self.methodB(1, 4)** |
| **22** | **def methodB(self, m, n):** |
| **23** | **y = self.temp + self.y + n** |
| **24** | **B.temp = m + self.y + n** |
| **25** | **self.methodA(n, m)** |
| **26** | **self.sum = self.y + y + A.temp** |
| **27** | **print(self.temp , y, self.sum)** |

**Illustrate** the output of the following statements:

**b1 = B()**

**b2 = B(b1)**

**Output:**

| **Out1** | **Out2** | **Out3** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  | **45** |
|  | **26** | **76** |

**Page 3 of 3 Set-A**