

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Insert your name, centre number and candidate number into the header above.

Save this evidence document as **evidence_** followed by your **centre number_ candidate number**.
For example: **evidence_ zz999_9999.doc**

Screenshots or program listings must be copied into appropriate cells in the following tables.

Examiners must be able to read the contents including any screenshots without the use of a magnifying glass. Answers that are not readable or missing will not be awarded any marks.

Save this evidence document at regular intervals, for example, every 10 minutes.

Question 1

Part 1(a)

```
def ReadData():
    try:
        # Open the file in read mode
        DataFile = open("Source files/Data.txt", "r")

        # Local array consist of 45 items
        DataItems = []

        # Read the data from the file and store it in the array
        for i in range(45):
            DataItems.append(DataFile.readline().strip())

        DataFile.close() # Close the file

        return DataItems
    except IOError:
        print("Error reading from file")
```

Part 1(b)(i)

```
# Function to format the array takes in an array of strings
def FormatArray(DataArray):
    OutputString = ""

    # Loop through the array
    for i in range(len(DataArray)):
        # Add the item to the output string with a space between each
        item
        OutputString = OutputString + DataArray[i] + " "

    return OutputString
```

Part 1(b)(ii)

```
FileData = ReadData() # Read the data from the file
FormattedArray = FormatArray(FileData) # Print the formatted data

print(FormattedArray) # Print the formatted data
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 1(b)(iii)

```
9618_w24_on_43 > python Question1_N24.py main
beige green scarlet silver bronze slate yellow orange jade lavender
magnolia magenta turquoise black grey russet maroon mango mint purpl
e red pink white cream navy olive brown violet cyan amber aqua azure
copper fawn fuschia gold indigo ivory mauve mulberry peach periwink
le plum rose sage
```

Part 1(c)

```
def CompareStrings(String1, String2):
    maxLen = len(String1) if len(String1) > len(String2) else
len(String2)

    for i in range(maxLen):
        if ord(String1[i]) < ord(String2[i]):
            return 1
        elif ord(String1[i]) > ord(String2[i]):
            return 2
        else:
            continue

    return -1
```

Part 1(d)(i)

```
def Bubble(DataArray):
    data = DataArray[:]
    n = len(data)

    for i in range(n):
        swapped = False
        for j in range(0, n-i-1):
            if CompareStrings(data[j], data[j + 1]) == 2:
                data[j], data[j + 1] = data[j + 1], data[j]
                swapped = True
        if not swapped:
            break

    return data
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 1(d)(ii)

```
SortedData = Bubble(FileData) # Sort the data
FormattedSortedData = FormatArray(SortedData) # Print the formatted data
print(FormattedSortedData) # Print the formatted data
```

Part 1(d)(iii)

```
9618_w24_on_43 > python Question1_N24.py main
amber aqua azure beige black bronze brown copper cream cyan fawn fuschia
gold green grey indigo ivory jade lavender magenta magnolia mango maroon
mauve mint mulberry navy olive orange peach periwinkle pink plum purple
red rose russet sage scarlet silver slate turquoise violet white yellow
```

Question 2

Part 2(a)(i)

```
class Horse:

    def __init__(self, Name, MaxFenceHeight, PercentageSuccess):
        self.__Name = Name # Private attribute
        self.__MaxFenceHeight = MaxFenceHeight # Private attribute
        self.__PercentageSuccess = PercentageSuccess # Private attribute
```

Part 2(a)(ii)

```
def GetName(self):
    return self.__Name

def GetMaxFenceHeight(self):
    return self.__MaxFenceHeight
```

Part 2(b)(i)

```
Horses = [] # Array of 2 horses object
Horses.append(Horse("Beauty", 150, 72))
Horses.append(Horse("Jet", 160, 65))

# Output the name of both horses from the array
print(Horses[0].GetName())
print(Horses[1].GetName())
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 2(b)(ii)

```
9618_w24_on_43 > python Question2_N24.py
Beauty
Jet
```

Part 2(c)(i)

```
class Fence:

    def __init__(self, Height, Risk):
        # Private attribute integer between 70 and 180
        self.__Height = Height
        # Private attribute integer between 1 and 5
        self.__Risk = Risk

    def GetHeight(self):
        return self.__Height

    def GetRisk(self):
        return self.__Risk
```

Part 2(c)(ii)

```
Courses = [] # Array of 4 fence object

for i in range(4):
    valid_input = False

    while not valid_input:
        # Read the height and risk of the fence
        height = int(input("Enter the height of the fence: "))
        risk = int(input("Enter the risk of the fence: "))

        if height < 70 or height > 180 or risk < 1 or risk > 5:
            print("Invalid input. Please enter a height between 70 and 180 and a risk between 1 and 5.")
        else:
            Courses.append(Fence(height, risk))
            print("Fence added. \n")
            valid_input = True
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 2(d)

```
def Success(self, FenceHeight, RiskModifier):
    if FenceHeight > self.__MaxFenceHeight:
        return self.__PercentageSuccess * 0.2
    else:
        if RiskModifier == 1:
            return self.__PercentageSuccess * 1
        elif RiskModifier == 2:
            return self.__PercentageSuccess * 0.9
        elif RiskModifier == 3:
            return self.__PercentageSuccess * 0.8
        elif RiskModifier == 4:
            return self.__PercentageSuccess * 0.7
        elif RiskModifier == 5:
            return self.__PercentageSuccess * 0.6
        else:
            return 0 # Invalid risk modifier
```

Part 2(e)(i)

```
for i in range(2):
    for j in range(4):
        print("The horse " + Horses[i].GetName() + " at fence " + str(j + 1) + " has a " + str(Horses[i].Success(Courses[j].GetHeight(), Courses[j].GetRisk())) + "% chance of success.")
```

Part 2(e)(ii)

```
average_success = [] # Array of average success rate for each horse

for i in range(2):
    total_success = 0
    for j in range(4):
        print("The horse " + Horses[i].GetName() + " at fence " + str(j + 1) + " has a " + str(Horses[i].Success(Courses[j].GetHeight(), Courses[j].GetRisk())) + "% chance of success.")
        total_success += Horses[i].Success(Courses[j].GetHeight(), Courses[j].GetRisk())

    print("The horse " + Horses[i].GetName() + " has an average " + str(total_success/4) + "%. chance of jumping over all four fences. \n")

    average_success.append(total_success/4)

if average_success[0] > average_success[1]:
    print("The horse " + Horses[0].GetName() + " has the highest average chance of success ")
elif average_success[0] < average_success[1]:
    print("The horse " + Horses[1].GetName() + " has the highest average chance of success ")
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 2(e)(iii)

```
9618_w24_on_43 > python Question2_N24.py
```

```
Beauty
```

```
Jet
```

```
Enter the height of the fence: 152
```

```
Enter the risk of the fence: 5
```

```
Fence added.
```

```
Enter the height of the fence: 121
```

```
Enter the risk of the fence: 1
```

```
Fence added.
```

```
Enter the height of the fence: 130
```

```
Enter the risk of the fence: 3
```

```
Fence added.
```

```
Enter the height of the fence: 145
```

```
Enter the risk of the fence: 4
```

```
Fence added.
```

```
The horse Beauty at fence 1 has a 14.4% chance of success.
```

```
The horse Beauty at fence 2 has a 72% chance of success.
```

```
The horse Beauty at fence 3 has a 57.6% chance of success.
```

```
The horse Beauty at fence 4 has a 50.4% chance of success.
```

```
The horse Beauty has an average 48.6%. chance of jumping over all four fences.
```

```
The horse Jet at fence 1 has a 39.0% chance of success.
```

```
The horse Jet at fence 2 has a 65% chance of success.
```

```
The horse Jet at fence 3 has a 52.0% chance of success.
```

```
The horse Jet at fence 4 has a 45.5% chance of success.
```

```
The horse Jet has an average 50.375%. chance of jumping over all four fences.
```

```
The horse Jet has the highest average chance of success
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Question 3

Part 3(a)

```
LinkedList = [] # Global array
FirstNode = - 1
FirstEmpty = 0

for i in range(20):
    LinkedList.append([-1, i + 1])

LinkedList[19][1] = -1
```

Part 3(b)

```
def InsertData():
    global FirstNode
    global FirstEmpty
    global LinkedList

    for i in range(5):
        if FirstEmpty != -1:
            NewNode = FirstEmpty
            FirstEmpty = LinkedList[FirstEmpty][1]
            LinkedList[NewNode][0] = int(input("Enter a number: "))
            LinkedList[NewNode][1] = FirstNode
            FirstNode = NewNode
```

Part 3(c)(i)

```
def OutputLinkedList():
    global FirstNode
    global LinkedList

    CurrentNode = FirstNode

    while CurrentNode != -1:
        print(LinkedList[CurrentNode][0])
        CurrentNode = LinkedList[CurrentNode][1]
```

Candidate Name: NGO THO QUANG

Centre Number: VN288

Candidate Number: 0012

Part 3(c)(ii)

```
InsertData()  
OutputLinkedList()
```

Part 3(c)(iii)

```
FirstEmpty: 0  
9618_w24_on_43 > python Question3_N24.py  
Enter a number: 5  
Enter a number: 1  
Enter a number: 2  
Enter a number: 3  
Enter a number: 8  
8  
3  
2  
1  
5
```

Part 3(d)(i)

```
def RemoveData(dataItem):  
    global FirstNode  
    global FirstEmpty  
    global LinkedList  
  
    CurrentNode = FirstNode  
    PreviousNode = -1  
  
    while CurrentNode != -1:  
        if LinkedList[CurrentNode][0] == dataItem:  
            if PreviousNode == -1:  
                FirstNode = LinkedList[CurrentNode][1]  
            else:  
                LinkedList[PreviousNode][1] = LinkedList[CurrentNode][1]  
            LinkedList[CurrentNode][1] = FirstEmpty  
            FirstEmpty = CurrentNode  
            return  
        PreviousNode = CurrentNode  
        CurrentNode = LinkedList[CurrentNode][1]
```


Candidate Name: NGO THO QUANG

Centre Number: VN288

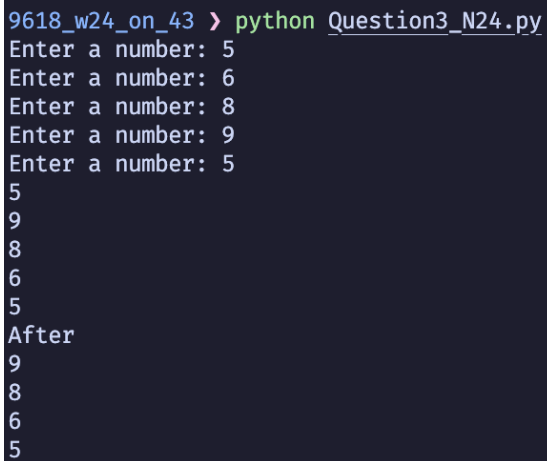
Candidate Number: 0012

Part 3(d)(ii)

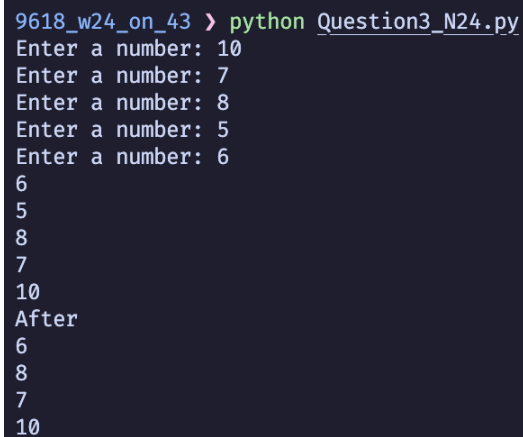
```
InsertData()  
OutputLinkedList()  
RemoveData(5)  
print("After")  
OutputLinkedList()
```

Part 3(d)(iii)

{Copy and paste the screenshot(s) for question 3(d)(iii) here}



```
9618_w24_on_43 > python Question3_N24.py  
Enter a number: 5  
Enter a number: 6  
Enter a number: 8  
Enter a number: 9  
Enter a number: 5  
5  
9  
8  
6  
5  
After  
9  
8  
6  
5
```



```
9618_w24_on_43 > python Question3_N24.py  
Enter a number: 10  
Enter a number: 7  
Enter a number: 8  
Enter a number: 5  
Enter a number: 6  
6  
5  
8  
7  
10  
After  
6  
8  
7  
10
```