

2(a)	1 mark each to max 7 • Function header (and end) • Prompt to enter filename and reading input • Opening the file (to read) and closing the file in an appropriate place • Looping until EOF ... • ... reading each line in the file ... • ... (removing line break and) inserting in array • Returning populated array • Exception handling try catch with appropriate output
2(b)	1 mark each • Procedure header (and end) taking (1D array) <code>dataArray</code> (of strings) as a parameter • Declaration/use of 6 1D arrays (equivalent), one for each colour • Looping through each line in parameter <code>dataArray</code> ... • ... splitting by comma • Comparing 2nd value/colour to each colour to select array ... • ... storing 1st value/integer in correct array

2(c)	1 mark each • Procedure header taking (1D) array and filename as parameters, opening file to append and closing file (in appropriate place) • Looping through each item in array parameter ... • ... writing to the file • ... with new line break between each line • Using exception handling try and catch with suitable output
------	--

Example program code:

Python

```
def StoreData(DataToStore, FileName):
    try:
        File = open(FileName, "a+")
        for Item in DataToStore:
            File.write(Item)
            File.write("\n")
        File.close()
    except:
        print("Cannot create or write to file")
```

VB.NET

```
Sub StoreData(DataToStore(), FileName)
    Dim FileWriter As IO.StreamWriter = New IO.StreamWriter(FileName, False)
    Dim x As Integer = 0
    Try
        While DataToStore(x) IsNot Nothing
            FileWriter.WriteLine(DataToStore(x))
            x = x + 1
        End While
        FileWriter.Close()
    Catch ex As Exception
        Console.WriteLine("Cannot open or write to file")
    End Try
End Sub
```

2(d)	1 mark each • Calling StoreData with one array and filename • Calling StoreData with remaining 5 arrays and filename
------	---

Example program code:

Python

```
StoreData(Red, "Red.txt")
StoreData(Green, "Green.txt")
StoreData(Blue, "Blue.txt")
StoreData(Orange, "Orange.txt")
StoreData(Yellow, "Yellow.txt")
StoreData(Pink, "Pink.txt")
```

VB.NET

```
StoreData(Red, "Red.txt")
StoreData(Green, "Green.txt")
StoreData(Blue, "Blue.txt")
StoreData(Orange, "Orange.txt")
StoreData(Yellow, "Yellow.txt")
StoreData(Pink, "Pink.txt")
```

Java

```
StoreData(Red, "Red.txt");
StoreData(Green, "Green.txt");
StoreData(Blue, "Blue.txt");
StoreData(Orange, "Orange.txt");
StoreData(Yellow, "Yellow.txt");
StoreData(Pink, "Pink.txt");
```

2(e)(i)	1 mark each • Calling <code>ReadData()</code> ... • ... and storing/using return value • Calling <code>SplitData()</code> with returned array as a parameter
Example program code: Python <pre>DataFromFile = ReadData() SplitData(DataFromFile)</pre> VB.NET <pre>Sub Main(args As String()) Dim DataFromFile(,) As String = ReadData() SplitData(DataFromFile) End Sub</pre> Java <pre>public static void main(String args[]){ String[][] DataFromFile = ReadData(); SplitData(DataFromFile); }</pre>	
2(e)(ii)	1 mark screenshots showing • Prompt and input of filename <code>TheData.txt</code> • Screenshot of data in red file. Filename must be shown in same screenshot as data