GradeMax

CAMBRIDGE
International Education

Cambridge International AS & A Level

COMPUTER SCIENCE **9618/42**

Paper 4 Practical **October/November 2025**

MARK SCHEME

Maximum Mark: 75

Published

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the October/November 2025 series for most Cambridge IGCSE, Cambridge International A and AS Level components, and some Cambridge O Level components.

© Cambridge University Press & Assessment 2025

GradeMax

PUBLISHED

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptions for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently, e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

© Cambridge University Press & Assessment 2025

GradeMax

GENERIC MARKING PRINCIPLE 5: Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).
GENERIC MARKING PRINCIPLE 6: Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

© Cambridge University Press & Assessment 2025

GradeMax

Annotations guidance for centres

Examiners use a system of annotations as a shorthand for communicating their marking decisions to one another. Examiners are trained during the standardisation process on how and when to use annotations. The purpose of annotations is to inform the standard isation and monitoring processes and guide the supervising examiners when they are checking the work of examiners within their team. The meaning of annotations and how they are used is specific to each component and is understood by all examiners who mark the component.

We publish annotations in our mark schemes to help centres understand the annotations they may see on copies of scripts. Note that there may not be a direct correlation between the number of annotations on a script and the mark awarded. Similarly, the use of an annotation may not be an indication of the quality of the response.

The annotations listed below were available to examiners marking this component in this series.

Annotations

Annotation	Meaning
	Benefit of the doubt
	To indicate where a key word/phrase/code is missing
	Incorrect
	Follow through
	Indicate a point in an answer
Highlighted text	To draw attention to a particular aspect or to indicate where parts of an answer have been combined
	Ignore
	Not answered question
	No examples or not enough

© Cambridge University Press & Assessment 2025

GradeMax

Annotation	Meaning
	Not relevant or used to separate parts of an answer
Off-page comment	Allows comments to be entered at the bottom of the RM marking window and then displayed when the associated question item is navigated to.
	Repetition
	Indicates that work or a page has been seen including blank answer spaces and blank pages.
	Correct
	Too vague

Mark scheme abbreviations

- **Bold** in mark scheme means that idea is required.
- / in mark scheme means alternative.
- // in mark scheme means alternative solution that gains the same mark point.
- … at the end of one mark point without a … at the start of the next just means the sentence follows on. There is no dependency.
- … at the end of one mark point and … at the start of the next, this means the second cannot be awarded without the first.
- () means what is in the brackets is not required, or it is not required in some languages but may be required in others.

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(a)(i)	1 mark each
	• Class header (and end)
	• 4 private attributes with correct data types
	• Constructor header (and end) taking 2 parameters within class …
	• … within constructor assigning parameters to `Species` and `DistancePerHour` and assigning 500.0 to `XPosition` and `YPosition`
	
	Example program code
	
	Java
	`class Bird{`
	`  private String Species;`
	`  private Double DistancePerHour;`
	`  private Double XPosition;`
	`  private Double YPosition;`
	
	`  public Bird(Double pDistancePerHour, String pSpecies){`
	`  Species = pSpecies;`
	`  DistancePerHour = pDistancePerHour;`
	`  XPosition = 500.0;`
	`  YPosition = 500.0;}}`
	
	VB.NET
	`Class Bird`
	`  Private Species As String`
	`  Private DistancePerHour As Single`
	`  Private XPosition As Single`
	`  Private YPosition As Single`
	`  Sub New(pDistancePerHour, pSpecies)`
	`  Species = pSpecies`
	`  DistancePerHour = pDistancePerHour`
	`  XPosition = 500.0`
	`  YPosition = 500.0`
	`  End Sub`
	`End Class`
		4

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(a)(i)	Python
	`class Bird:`
	`  def __init__(self, pDistancePerHour, pSpecies):`
	`  self.__Species = pSpecies #string`
	`  self.__DistancePerHour = pDistancePerHour #real`
	`  self.__XPosition = 500.0 #real`
	`  self.__YPosition = 500.0 #real`	

1(a)(ii)	1 mark each
	• Get method header (and end) with no parameter ….
	• … returning `Species`
	
	Example program code
	
	Java
	`public String GetSpecies(){`
	`  return Species;`
	`}`
	
	VB.NET
	`Function GetSpecies()`
	`  Return Species`
	`End Function`
	
	Python
	`def GetSpecies(self):`
	`  return self.__Species`
		2

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(a)(iii)	1 mark each
	• Get method header (and end) with no parameter, returning a value
	• Creating **correct** string using **attributes** …
	• … returning this string
	
	Example program code
	
	Java
	`public String GetPosition(){`
	`  String ReturnValue = "X = " + XPosition + " Y = " + YPosition;`
	`  return ReturnValue;`
	`}`
	
	VB.NET
	`Function GetPosition()`
	`  Dim ReturnValue As String = "X = " & XPosition & " Y = " & YPosition`
	`  Return ReturnValue`
	`End Function`
	
	Python
	`def GetPosition(self):`
	`  ReturnValue = "X = " + str(self.__XPosition) + " Y = " + str(self.__YPosition)`
	`  return ReturnValue`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(a)(iv)	1 mark each
	• Method header (and end) taking direction and minutes flying as parameters
	• Calculation of distance using minutes flying parameter and attribute `DistancePerHour`
	• Selection based on direction parameter …
	• … north adding to `YPosition` **and** south subtracting from `YPosition`
	• … east adding to `XPosition` **and** west subtracting from `XPosition`
	
	Example program code
	
	Java
	`public Boolean Move(String Direction, Integer MinsFlying){`
	`  if(Direction.compareTo("E") == 0){`
	`  XPosition = XPosition + ((DistancePerHour / 60) * MinsFlying);`
	
	`  }else if(Direction.compareTo("W") == 0){`
	`  XPosition = XPosition - ((DistancePerHour / 60) * MinsFlying);`
	
	`  }else if(Direction.compareTo("N") == 0){`
	`  YPosition = YPosition + ((DistancePerHour / 60) * MinsFlying);`
	
	`  }else if(Direction.compareTo("S") == 0){`
	`  YPosition = YPosition - ((DistancePerHour / 60) * MinsFlying);`
	
	`  }`
	`}`
		5

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(a)(iv)	VB.NET
	`Function Move(Direction, MinsFlying)`
	`    If Direction = "E" Then`
	`        XPosition = XPosition + ((DistancePerHour / 60) * MinsFlying)`
	
	`    ElseIf Direction = "W" Then`
	`        XPosition = XPosition - ((DistancePerHour / 60) * MinsFlying)`
	
	`    ElseIf Direction = "N" Then`
	`        YPosition = YPosition + ((DistancePerHour / 60) * MinsFlying)`
	
	`    ElseIf Direction = "S" Then`
	`        YPosition = YPosition - ((DistancePerHour / 60) * MinsFlying)`
	
	`    End If`
	`End Function`
	
	Python
	`def Move(self, Direction, MinsFlying):`
	`    if Direction == "E":`
	`        self.__XPosition = self.__XPosition + ((self.__DistancePerHour/60)*MinsFlying)`
	
	`    elif Direction == "W":`
	`        self.__XPosition = self.__XPosition - ((self.__DistancePerHour/60)*MinsFlying)`
	
	`    elif Direction == "N":`
	`        self.__YPosition = self.__YPosition + ((self.__DistancePerHour/60)*MinsFlying)`
	
	`    elif Direction == "S":`
	`        self.__YPosition = self.__YPosition - ((self.__DistancePerHour/60)*MinsFlying)`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(b)	1 mark each
	• Cockatiel 71.0 instance of `Bird` created …
	• Macaw 56.0 instance of `Bird` created …
	• … both stored in variables/structures
	
	Example program code
	
	Java
	`Bird FirstBird = new Bird(71.0, "Cockatiel");`
	`Bird SecondBird = new Bird(56.0, "Macaw");`
	
	VB.NET
	`Dim FirstBird As Bird = New Bird(71.0, "Cockatiel")`
	`Dim SecondBird As Bird = New Bird(56.0, "Macaw")`
	
	Python
	`FirstBird = Bird(71.0, "Cockatiel")`
	`SecondBird = Bird(56.0, "Macaw")`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(c)(i)	1 mark each
	• Output species, X and Y position for both `Bird` objects in appropriate messages(s)
	• Prompt **and** input of bird choice, time and direction …
	• … validating all **three** inputs …
	• … looping until all **three** are valid
	• Calling `Move()` for chosen bird …
	• … with **only** input direction and input time as arguments
	• Outputting new position for the bird moved
	• Using get methods throughout where appropriate
	
	Example program code
	
	Java
	`Integer Choice;`
	`Integer Time;`
	`String Direction;`
	`Scanner scanner = new Scanner(System.in);`
	
	`Choice = 0;`
	`while(Choice != 1 && Choice != 2){`
	`System.out.println("Which bird do you want to move");`
	`System.out.println("Enter 1 for " + FirstBird.GetSpecies() + " is currently at " + FirstBird.GetPosition());`
	`System.out.println("Enter 2 for " + SecondBird.GetSpecies() + " is currently at " + SecondBird.GetPosition());`
	`Choice = Integer.parseInt(scanner.nextLine());`
	`}`
	
	`Time = -1;`
	`while(Time < 0 or Time > 500){`
	`System.out.println("To the nearest minute how long as the bird been flying?");`
	`Time = Integer.parseInt(scanner.nextLine());`
	`}`
	`Boolean Valid = false;`
	`while(Valid == false){`
		8

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(c)(i)	Valid = true;
	System.out.println("Which direction has the bird been flying, North, South, East or West?");
	Direction = scanner.nextLine().toUpperCase();
	if(Direction.compareTo("NORTH") == 0 || Direction.compareTo("N") == 0){
	if(Choice == 1){
	FirstBird.Move("N",Time);
	}else{
	SecondBird.Move("N", Time);
	}
	} else if(Direction.compareTo("SOUTH")== 0 || Direction.compareTo("S") == 0){
	if(Choice == 1){
	FirstBird.Move("S",Time);
	}else{
	SecondBird.Move("S", Time);
	}
	} else if(Direction.compareTo("EAST")== 0 || Direction.compareTo("E") == 0){
	if(Choice == 1){
	FirstBird.Move("E",Time);
	}else{
	SecondBird.Move("E", Time);
	}
	} else if(Direction.compareTo("WEST")== 0 || Direction.compareTo("W") == 0){
	if(Choice == 1){
	FirstBird.Move("W",Time);
	}else{
	SecondBird.Move("W", Time);
	}
	
	}else{
	Valid = false;
	}
	
	}
	System.out.println(FirstBird.GetSpecies() + " is currently at " + FirstBird.GetPosition());	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(c)(i)	`System.out.println(SecondBird.GetSpecies() + " is currently at " + SecondBird.GetPosition());`
	
	VB.NET
	`Dim Choice As Integer`
	`Dim Time As Integer`
	`Dim Direction As String`
	
	`Choice = 0`
	
	`While Choice <> 1 And Choice <> 2`
	`  Console.WriteLine("Which bird do you want to move")`
	`  Console.WriteLine("Enter 1 for " & FirstBird.GetSpecies() & " is currently at " & FirstBird.GetPosition())`
	`  Console.WriteLine("Enter 2 for " & SecondBird.GetSpecies() & " is currently at " & SecondBird.GetPosition())`
	`  Choice = Console.ReadLine`
	`End While`
	`Time = -1`
	`While Time < 0 Or Time > 500`
	`  Console.WriteLine("To the nearest minute how long has the bird been flying ")`
	`  Time = Console.ReadLine`
	`End While`
	`Dim Valid As Boolean = False`
	`While (Valid = False)`
	`  Valid = True`
	
	`  Console.WriteLine("Which direction has the bird been flying, North, South, East or West ")`
	`  Direction = Console.ReadLine().ToUpper`
	`  If Direction = "NORTH" Or Direction = "N" Then`
	`  If Choice = 1 Then`
	`  FirstBird.Move("N", Time)`
	`  Else`
	`  SecondBird.Move("N", Time)`
	`  End If`	

© Cambridge University Press & Assessment 2025

Question	Answer	Marks

1(c)(i)	ElseIf Direction = "SOUTH" Or Direction = "S" Then
	If Choice = 1 Then
	FirstBird.Move("S", Time)
	Else
	SecondBird.Move("S", Time)
	End If
	ElseIf Direction = "EAST" Or Direction = "E" Then
	If Choice = 1 Then
	FirstBird.Move("E", Time)
	Else
	SecondBird.Move("E", Time)
	End If
	ElseIf Direction = "WEST" Or Direction = "W" Then
	If Choice = 1 Then
	FirstBird.Move("W", Time)
	Else
	SecondBird.Move("W", Time)
	End If
	Else
	Valid = False
	End If
	
	End While
	Console.WriteLine(FirstBird.GetSpecies() & " is currently at " & FirstBird.GetPosition())
	Console.WriteLine(SecondBird.GetSpecies() & " is currently at " & SecondBird.GetPosition())
	
	Python
	Choice = 0
	while Choice != 1 and Choice != 2:
	print("Which bird do you want to move")
	print("Enter 1 for", FirstBird.GetSpecies(), "is currently at",
	FirstBird.GetPosition())
	Choice = -1
	
	print("Enter 2 for", SecondBird.GetSpecies(), "is currently at",
	SecondBird.GetPosition())	

GradeMax

Question	Answer	Marks

1(c)(i)	`    Choice = int(input())`
	`Time = -1`
	`while Time < 0 || Time > 500:`
	`    Time = int(input("To the nearest minute how long has the bird been flying "))`
	`    Valid = False`
	`while Valid == False:`
	`    Valid = True`
	`    Direction = input("Which direction has the bird been flying, North, South, East or West `
	`").upper()`
	`    if Direction == "NORTH" or Direction == "N":`
	`        if Choice == 1:`
	`            FirstBird.Move("N",Time)`
	`        else:`
	`            SecondBird.Move("N",Time)`
	`    elif Direction == "SOUTH" or Direction == "S":`
	`        if Choice == 1:`
	`            FirstBird.Move("S",Time)`
	`        else:`
	`            SecondBird.Move("S",Time)`
	`    elif Direction == "EAST" or Direction == "E":`
	`        if Choice == 1:`
	`            FirstBird.Move("E",Time)`
	`        else:`
	`            SecondBird.Move("E",Time)`
	`    elif Direction == "WEST" or Direction == "W":`
	`        if Choice == 1:`
	`            FirstBird.Move("W",Time)`
	`        else:`
	`            SecondBird.Move("W",Time)`
	`    else:`
	`        Valid = False`
	
	`print(FirstBird.GetSpecies(), "is currently at", FirstBird.GetPosition())`
	`print(SecondBird.GetSpecies(), "is currently at", SecondBird.GetPosition())`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

1(c)(ii)	1 mark for each:
	• screenshot showing inputs for one test with correct output
	• screenshot showing inputs for a second test with correct output
	• screenshot showing inputs for a third and fourth test with correct output
	
	e.g.
	
	Test 1:
	`Which bird do you want to move`
	`Enter 1 for Cockatiel is currently at X = 500.0 Y = 500.0`
	`Enter 2 for Macaw is currently at X = 500.0 Y = 500.0`
	`1`
	`To the nearest minute how long has the bird been flying 60`
	`Which direction has the bird been flying, North, South, East or West N`
	`Cockatiel is currently at X = 500.0 Y = 571.0`
	`Macaw is currently at X = 500.0 Y = 500.0`
	
	Test 2:
	`Which bird do you want to move`
	`Enter 1 for Cockatiel is currently at X = 500.0 Y = 500.0`
	`Enter 2 for Macaw is currently at X = 500.0 Y = 500.0`
	`2`
	`To the nearest minute how long has the bird been flying 30`
	`Which direction has the bird been flying, North, South, East or West S`
	`Cockatiel is currently at X = 500.0 Y = 500.0`
	`Macaw is currently at X = 500.0 Y = 472.0`
	
	Test 3:
	`Which bird do you want to move`
	`Enter 1 for Cockatiel is currently at X = 500.0 Y = 500.0`
	`Enter 2 for Macaw is currently at X = 500.0 Y = 500.0`
	`1`
	`To the nearest minute how long has the bird been flying 30`
	`Which direction has the bird been flying, North, South, East or West W`
	`Cockatiel is currently at X = 464.5 Y = 500.0`
	`Macaw is currently at X = 500.0 Y = 500.0`
	
	Test 4:
	`Which bird do you want to move`
	`Enter 1 for Cockatiel is currently at X = 500.0 Y = 500.0`
	`Enter 2 for Macaw is currently at X = 500.0 Y = 500.0`
	`2`
	`To the nearest minute how long has the bird been flying 60`
	`Which direction has the bird been flying, North, South, East or West E`
	`Cockatiel is currently at X = 500.0 Y = 500.0`
	`Macaw is currently at X = 556.0 Y = 500.0`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(a)	1 mark each
	• Creation of 1D array …
	• … with 20 generated random integers between 0 and 100 (inclusive) …
	• … all 20 random integers are unique
	
	Example program code
	
	Java
	`public static void main(String args[]){`
	`  Integer[] TheArray = new Integer[20];`
	`  Integer Generated;`
	`  Integer X = 0;`
	`  Random RandomNumber = new Random();`
	`  while(X < 20){`
	`  Generated = RandomNumber.nextInt(101);`
	`  if(Arrays.asList(TheArray).indexOf(Generated) < 0){`
	`  TheArray[X] = Generated;`
	`  X++;`
	`  }`
	`  }`
	`}`
	
	VB.NET
	`Dim RandomNumber As Random = New Random()`
	`Dim TheArray(19) As Integer`
	`Dim Generated As Integer`
	`Dim X As Integer = 0`
	
	`While X < 20`
	`  Generated = RandomNumber.Next(0, 100)`
	`  If Array.IndexOf(TheArray, Generated) < 0 Then`
	`  TheArray(X) = Generated`
	`  X = X + 1`
	`  End If`
	`End While`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(a)	Python
	`TheArray = []`
	`TheArray = random.sample(range(0,101),20)`	

2(b)	1 mark each
	• Procedure header (and close) taking (array) as parameter
	• Outputting array contents once …
	• … on one line with a space between each integer
	
	Example program code
	
	Java
	`public static void PrintArray(Integer[] DataArray){`
	`  String Output = "";`
	`  for(Integer X = 0; X < 20; X++){`
	`  Output += Integer.toString(DataArray[X]) + " ";`
	`  }`
	`  System.out.println(Output);`
	`}`
	
	VB.NET
	`Sub PrintArray(DataArray() As Integer)`
	`  Dim Output As String = ""`
	`  For X = 0 To 19`
	`  Output = Output + Str(DataArray(X)) + " "`
	`  Next X`
	`  Console.WriteLine(Output)`
	`End Sub`
	
	Python
	`def PrintArray(DataArray):`
	`  Output = ""`
	`  for Item in DataArray:`
	`  Output = Output + str(Item) + " "`
	`  print(Output)`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(c)	1 mark each
	• Function header (and end) taking (array) parameter and returning a sorted array after sorting
	• Outer loop …
	• … inner loop ...
	• … comparing elements and swapping into ascending order
	• Sort must work for array of any length i.e. loops for length of parameter array
	
	Example program code
	
	Java
	`public static Integer[] BubbleSort(Integer[] DataArray){`
	`  Boolean Swap = true;`
	`  Integer Temp;`
	`  while(Swap){`
	`  Swap = false;`
	`  for(Integer X = 0; X < DataArray.length - 1; X++){`
	`  if(DataArray[X] > DataArray[X+1]){`
	`  Temp = DataArray[X];`
	`  DataArray[X] = DataArray[X + 1];`
	`  DataArray[X + 1] = Temp;`
	`  Swap = true;`
	`  }`
	`  }`
	`  }`
	`  return DataArray;`
	`}`
		5

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(c)	VB.NET
	`Function BubbleSort(DataArray() As Integer)`
	`  Dim Swap As Boolean = True`
	`  Dim Temp As Integer`
	`  While Swap = True`
	`  Swap = False`
	`  For X = 0 To DataArray.Length - 2`
	`  If DataArray(X) > DataArray(X + 1) Then`
	`  Temp = DataArray(X)`
	`  DataArray(X) = DataArray(X + 1)`
	`  DataArray(X + 1) = Temp`
	
	`  Swap = True`
	`  End If`
	`  Next X`
	`  End While`
	
	`  Return DataArray`
	`End Function`
	
	Python
	`def BubbleSort(DataArray):`
	`  Swap = True`
	
	`  while Swap  == True:`
	`  Swap = False`
	`  for y in range(0, len(DataArray)-1):`
	`  if DataArray[y] > DataArray[y+1]:`
	`  DataArray[y], DataArray[y+1] = DataArray[y+1], DataArray[y]`
	`  Swap = True`
	`  return DataArray`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(d)(i)	1 mark each
	• Calling `PrintArray()` with array as argument
	• Calling `BubbleSort()` with array as argument **and** storing/using return value
	• Outputting `"Sorted"` **and** calling `PrintArray()` with (returned) array as argument
	
	Example program code
	
	Java
	`PrintArray(TheArray);`
	`Integer[] SortedArray = new Integer[20];`
	`SortedArray = BubbleSort(TheArray);`
	`System.out.println("Sorted");`
	`PrintArray(SortedArray);`
	
	VB.NET
	`PrintArray(TheArray)`
	`Dim SortedArray(19) As Integer`
	`SortedArray = BubbleSort(TheArray)`
	`Console.WriteLine("Sorted")`
	`PrintArray(SortedArray)`
	
	Python
	`PrintArray(TheArray)`
	`SortedArray = BubbleSort(TheArray)`
	`print("Sorted")`
	`PrintArray(SortedArray)`
		3

2(d)(ii)	1 mark • Output shows unsorted array of 20 integers between 0 and 100 inclusive before sorting, “Sorted” output, array of the same integers sorting into ascending order. All screenshots will be unique to the candidate	**1**

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(e)	1 mark each
	• Function header (and end) taking four parameters **and** recursive function written
	• Calculating middle value
	• Comparing middle value to data parameter **and** returning index if equal
	• If middle is greater than, recursive call with middle –1 for upper
	• If middle is less than, recursive call with middle + 1 for lower
	• Checking if not found **and** returning –1
	
	Example program code
	
	Java
	`public static Integer RecursiveBinarySearch(Integer[] DataArray, Integer Lower, Integer Upper, Integer DataToFind){`
	`  Integer Middle;`
	`  if(Upper >= Lower){`
	`  Middle = Lower + (Upper - Lower) / 2;`
	`  if(DataArray[Middle] == DataToFind){`
	`  return Middle;`
	`  }else if(DataArray[Middle] > DataToFind){`
	`  return RecursiveBinarySearch(DataArray, Lower, Middle - 1, DataToFind);`
	`  }else{`
	`  return RecursiveBinarySearch(DataArray, Middle + 1, Upper, DataToFind);`
	`  }`
	`  }else{`
	`  return -1;`
	`  }`
	`}`
		6

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(e)	VB.NET
	`Function RecursiveBinarySearch(DataArray() As Integer, Lower As Integer, Upper As Integer, DataToFind As Integer)`
	`    Dim Middle As Integer`
	`    If Upper >= Lower Then`
	`        Middle = Lower + (Upper - Lower) \ 2`
	`        If DataArray(Middle) = DataToFind Then`
	`            Return Middle`
	`        ElseIf DataArray(Middle) > DataToFind Then`
	`            Return RecursiveBinarySearch(DataArray, Lower, Middle - 1, DataToFind)`
	`        Else`
	`            Return RecursiveBinarySearch(DataArray, Middle + 1, Upper, DataToFind)`
	`        End If`
	`    Else`
	`        Return -1`
	`    End If`
	`End Function`
	
	Python
	`def RecursiveBinarySearch(DataArray, Lower, Upper, DataToFind):`
	`    if Upper >= Lower:`
	`        Middle = Lower + (Upper - Lower) // 2`
	`        if DataArray[Middle] == DataToFind:`
	`            return Middle`
	`        elif DataArray[Middle] > DataToFind:`
	`            return RecursiveBinarySearch(DataArray, Lower, Middle - 1, DataToFind)`
	`        else:`
	`            return RecursiveBinarySearch(DataArray, Middle + 1, Upper, DataToFind)`
	`    else:`
	`        return -1`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

2(f)(i)	1 mark each
	• Prompt and input of integer
	• Call of `RecursiveBinarySearch(SortedArray, 0, 19, input)` **and** storing/using return value
	• Output of `"Not found"` if `-1` returned **and** output `"Found at position"` with index returned if found
	
	Example program code
	
	Java
	`System.out.println("Enter the number to find");`
	`Scanner scanner = new Scanner(System.in);`
	`Integer DataToFind = Integer.parseInt(scanner.nextLine());`
	`Integer Location = RecursiveBinarySearch(SortedArray, 0, 19, DataToFind);`
	`if(Location == -1){`
	`  System.out.println("Not found");`
	`}else{`
	`  System.out.println("Found at position " + Location);`
	`}`
	
	VB.NET
	`Console.WriteLine("Enter the number to find ")`
	`Dim DataToFind As Integer = Console.ReadLine()`
	`Dim Location As Integer = RecursiveBinarySearch(SortedArray, 0, 19, DataToFind)`
	`If Location = -1 Then`
	`  Console.WriteLine("Not found")`
	`Else`
	`  Console.WriteLine("Found at position " & Location)`
	`End If`
	
	Python
	`DataToFind = int(input("Enter the number to find "))`
	`Location = RecursiveBinarySearch(SortedArray, 0, 19, DataToFind)`
	`if Location == -1:`
	`  print("Not found")`
	`else:`
	`  print("Found at position", Location)`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks
2(f)(ii)	1 mark each • screenshot showing smallest number in array input and found message with index 0 **and** screenshot showing highest number in array input and found message with index 19 • screenshot showing a number not in the array input **and** an output of `"Not found"` All screenshots will be unique to the candidate	**2**

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(a)	1 mark each
	• (global) `TreeArray` declared as a 2D array with 50 × 3 elements …
	• … all initialised to `-1`
	• (global) `RootPointer` initialised to `-1` **and** `FreeNode` initialised to 0
	
	Example program code
	
	Java
	`public static Integer FreeNode;`
	`public static Integer RootPointer;`
	`public static Integer[][] TreeArray = new Integer[50][3];`
	`public static void main(String args[]){`
	`  for(Integer X = 0; X < 50; X++){`
	`  TreeArray[X][0] = -1;`
	`  TreeArray[X][1] = -1;`
	`  TreeArray[X][2] = -1;`
	`  }`
	`  RootPointer = -1;`
	`  FreeNode = 0;`
	`}`
	
	VB.NET
	`Dim FreeNode As Integer`
	`Dim TreeArray(0 To 49, 0 To 2) As Integer`
	`Dim RootPointer As Integer`
	`Sub Main(args As String())`
	`  For X = 0 To 49`
	`  TreeArray(X, 0) = -1`
	`  TreeArray(X, 1) = -1`
	`  TreeArray(X, 2) = -1`
	`  Next`
	`  RootPointer = -1`
	`  FreeNode = 0`
	`End Sub`
		3

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(a)	Python
	`TreeArray = []`
	`for x in range(50):`
	`  TreeArray.append([-1,-1,-1])`
	`RootPointer = -1`
	`FreeNode = 0`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(b)	1 mark each to max 7
	• Procedure header (and end) taking one (integer) parameter **and** storing parameter in array in index `TreeArray[FreeNode][1]`
	• Checking if tree is full (`FreeNode = 50`) **and** outputting `"The tree is full"`
	• Checking if tree is empty (`RootPointer = -1 // FreeNode = 0`) **and** if so, storing 0 in `RootPointer`
	
	• (if not empty) Comparing parameter to data at index `TreeArray[RootPointer][1]` …
	• … if less than, accessing left node …
	• … if greater than, accessing right node …
	• … until location found …
	• … updating parent node's appropriate pointer
	
	• Incrementing `FreeNode`
	
	Example program code
	
	Java
	`public static void AddNode(Integer NodeData){`
	`  Boolean Placed;`
	`  Integer CurrentNode;`
	`  if(FreeNode <= 49){`
	`  TreeArray[FreeNode][0] = -1;`
	`  TreeArray[FreeNode][1] = NodeData;`
	`  TreeArray[FreeNode][2] = -1;`
	`  if(RootPointer == -1){`
	`  RootPointer = 0;`
	`  }else{`
	`  Placed = false;`
	`  CurrentNode = RootPointer;`
	`  while(Placed == false){`
	`  if(NodeData < TreeArray[CurrentNode][1]){`
	`  if(TreeArray[CurrentNode][0] == -1){`
	`  TreeArray[CurrentNode][0] = FreeNode;`
	`  Placed = true;`
	`  }else{`
		7

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks
3(b)		

```
                                    CurrentNode = TreeArray[CurrentNode][0];
                              }
                        }else{
                              if(TreeArray[CurrentNode][2] == -1){
                                    TreeArray[CurrentNode][2] = FreeNode;
                                    Placed = true;
                              }else{
                                    CurrentNode = TreeArray[CurrentNode][2];
                              }
                        }
                  }
            }
            FreeNode++;
      }else{
            System.out.println("The tree is full");
      }
}
```

VB.NET

```
Sub AddNode(NodeData)
    Dim Placed As Boolean
    Dim CurrentNode As Integer
    If FreeNode <= 49 Then
        TreeArray(FreeNode, 0) = -1
        TreeArray(FreeNode, 1) = NodeData
        TreeArray(FreeNode, 2) = -1

        If RootPointer = -1 Then
            RootPointer = 0
        Else
            Placed = False
            CurrentNode = RootPointer
        While Placed = False
            If NodeData < TreeArray(CurrentNode, 1) Then
                    If TreeArray(CurrentNode, 0) = -1 Then
                        TreeArray(CurrentNode, 0) = FreeNode
```

© Cambridge University Press & Assessment 2025

Question	Answer	Marks
3(b)		

```
                        Placed = True
                    Else
                        CurrentNode = TreeArray(CurrentNode, 0)
                    End If
                Else
                    If TreeArray(CurrentNode, 2) = -1 Then
                        TreeArray(CurrentNode, 2) = FreeNode
                        Placed = True
                    Else
                        CurrentNode = TreeArray(CurrentNode, 2)
                    End If
                End If
            End While
        End If
        FreeNode = FreeNode + 1
    Else
        Console.WriteLine("The tree is full")
    End If
End Sub
```

Python

```
def AddNode(NodeData):
    global FreeNode
    global TreeArray
    global RootPointer
    if FreeNode <= 49:
        TreeArray[FreeNode][0] = -1
        TreeArray[FreeNode][1] = NodeData
        TreeArray[FreeNode][2] = -1
        if RootPointer == -1:
            RootPointer = 0
        else:
            Placed = False
            CurrentNode = RootPointer
            while Placed == False:
                if NodeData < TreeArray[CurrentNode][1]:
```

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(b)	`if TreeArray[CurrentNode][0] == -1:`
	`  TreeArray[CurrentNode][0] = FreeNode`
	`  Placed = True`
	`else:`
	`  CurrentNode = TreeArray[CurrentNode][0]`
	`else:`
	`if TreeArray[CurrentNode][2] == -1:`
	`  TreeArray[CurrentNode][2] = FreeNode`
	`  Placed = True`
	`else:`
	`  CurrentNode = TreeArray[CurrentNode][2]`
	`FreeNode = FreeNode + 1`
	`else:`
	`print("The tree is full")`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(c)	1 mark each to max 4
	• Opening the file to read **and** closing the file in an appropriate place
	• Looping 50 times/through file/through each line/until EOF …
	• … reading in **each line** …
	• … calling `AddNode()` with each value read in
	• Exception handling try, catch, except with appropriate message
	
	Example program code
	
	Java
	`Integer Line;`
	`String ReadData;`
	`try{`
	`  FileReader f = new FileReader("TreeData.txt");`
	`  try{`
	`  BufferedReader Reader = new BufferedReader(f);`
	`  ReadData = Reader.readLine();`
	`  while (ReadData != null){`
	`  Line = Integer.parseInt(ReadData);`
	`  AddNode(Line);`
	`  ReadData = Reader.readLine();`
	`  }`
	`  Reader.close();`
	`  }catch(IOException ex){`
	`  }`
	`}catch(FileNotFoundException e){`
	`  System.out.println("File not found");`
	`}`
		4

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(c)	VB.NET
	`Try`
	`  Dim FileReader As New System.IO.StreamReader("TreeData.txt")`
	`  While Not FileReader.EndOfStream`
	`  AddNode(FileReader.ReadLine())`
	`  End While`
	`  FileReader.Close()`
	`Catch ex As Exception`
	`  Console.WriteLine("Cannot open file")`
	`End Try`
	
	Python
	`try:`
	`  File= open("TreeData.txt")`
	`  for Line in File:`
	`  AddNode(int(Line.strip()))`
	`  File.close()`
	`except:`
	`  print("Error cannot open file")`	

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(d)	1 mark each
	• Procedure header (and end) **and** with exception handling for writing to file: try, catch, except with appropriate message
	• Opening the file (`Tree.txt`) to write **and** closing the file in appropriate place
	• Looping through each element in array …
	• … creating correct string
	• … writing each string to the file
	
	Example program code
	
	Java
	`public static void WriteAllToFile(){`
	`  File TheFile = new File("Tree.txt");`
	`  String Line;`
	`  try{`
	`  FileWriter FW = new FileWriter(TheFile, true);`
	`  for(Integer X = 0; X < 50; X++){`
	`  Line = TreeArray[X][0] + "," + TreeArray[X][1] + "," + TreeArray[X][2];`
	`  FW.write(Line);`
	`  FW.write("\n");`
	`  }`
	`  FW.close();`
	`  }catch(IOException ex){`
	`  System.out.println("Cannot open file");`
	`  }`
	`}`
		5

© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(d)	VB.NET
	`Sub WriteAllToFile()`
	`  Dim FileWriter As IO.StreamWriter = New IO.StreamWriter("Tree.txt", False)`
	`  Dim Line As String`
	`  Try`
	`  For x = 0 To 49`
	`  Line = TreeArray(x, 0) & "," & TreeArray(x, 1) & "," & TreeArray(x, 2)`
	`  FileWriter.WriteLine(Line)`
	`  Next`
	`  FileWriter.Close()`
	`  Catch ex As Exception`
	`  Console.WriteLine("Cannot open or write to file")`
	`  End Try`
	`End Sub`
	
	Python
	`def WriteAllToFile():`
	`  try:`
	`  File = open("Tree.txt","a+")`
	`  for x in range(0, 50):`
	`  Line = str(TreeArray[x][0]) + "," + str(TreeArray[x][1])+ "," +`
	`str(TreeArray[x][2]) + "\n"`
	`  File.write(Line)`
	`  File.close()`
	`  except:`
	`  print("Cannot write to file")`	

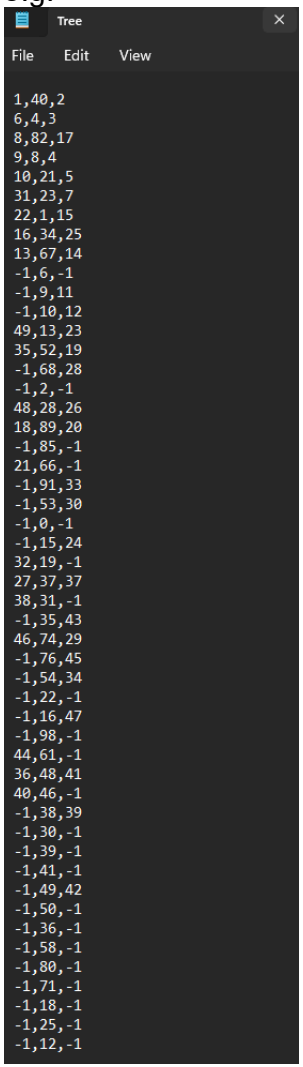
© Cambridge University Press & Assessment 2025

GradeMax

Question	Answer	Marks

3(e)(i)	1 mark for calling `WriteAllToFile()`
	
	Example program code
	
	Java
	`WriteAllToFile();`
	
	VB.NET
	`WriteAllToFile()`
	
	Python
	`WriteAllToFile()`
		1

© Cambridge University Press & Assessment 2025

Question	Answer	Marks
3(e)(ii)	1 mark for a screenshot that shows correct data stored, each node on a new line (in correct format). The screenshot must include the filename. e.g.  	**1**