



--	--	--

# 程序代写代做 CS编程辅导

Semester One 2018



Examination Period

Information Technology

EXAM CODES:

TITLE OF PAPER:

Computer Science - PAPER 1

EXAM DURATION:

2 hours writing time

READING AND NOTING TIME:

30 minutes

**THIS PAPER IS FOR STUDENTS STUDYING AT: (tick where applicable)**

- ☐ Caulfield
 ☒ Clayton
 ☐ Parkville
 ☐ Peninsula  
☐ Monash Extension
 ☐ Off Campus Learning
 ☒ Malaysia
 ☐ Sth Africa  
☐ Other (specify)

During an exam, you must not have in your possession any item/material that has not been authorised for your exam. This includes books, notes, paper, electronic device/s, mobile phone, smart watch/device, calculator, pencil case, or writing on any part of your body. Any authorised items are listed below. Items/materials on your desk, chair, in your clothing or otherwise on your person will be deemed to be in your possession.

**No examination materials are to be removed from the room.** This includes retaining, copying, memorising or noting down content of exam material for personal use or to share with any other person by any means following your exam.

Failure to comply with the above instructions, or attempting to cheat or cheating in an exam is a discipline offence under Part 7 of the Monash University (Council) Regulations.

## AUTHORISED MATERIALS

OPEN BOOK

☐ YES ☒ NO

CALCULATORS

☐ YES ☒ NO

SPECIFICALLY PERMITTED ITEMS

☐ YES ☒ NO

if yes, items permitted are:

Page	Marks	Page	Marks
3	10	23	3
5	3	25	8
7	8	29	5
9	3	31	6
11	6	35	10
13	9	37	8
15	8	<b>Total</b>	100
17	7		
21	6		

*Candidates must complete this section if required to write answers within this paper*

STUDENT ID: \_\_\_\_\_

DESK NUMBER: \_\_\_\_\_

# 程序代写代做 CS编程辅导

Question 1 [10 marks]

This question is about MIPS programming and function calls. Translate the following Python code faithfully into MIPS. Make sure you follow the MIPS function calling and memory usage conventions in the lectures. Use only instructions in the MIPS reference sheet. There is no code calling the function, thus the answer should involve only instructions by the callee.



Python	MIPS Code
def my_function(r):	
if n == 0:	
return m	
else:	
return m//n	

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

## 程序代写代做 CS编程辅导

### Question 2 – Array-based structures [11 marks]

This question is about Array-based structures. The partial implementation below is from a Queue whose underlying array is automatically resizable. The Queue uses the space of the array efficiently, wrapping around the front and the rear indices (i.e. a circular queue). In a Queue, the size of the underlying array when appending to a Queue that is a full space available. The partial implementation is as follows:



```
1 class Queue:
2
3     def __init__(self):
4         self.array = build_array(10)
5         self.front = 0
6         self.rear = 0
7         self.count = 0
8
9     def is_full(self):
10         return False
11
12     def is_empty(self):
13         return self.count == 0
14
15     def __len__(self):
16         return self.count
17
18     def append(self, new_item):
19         if self.count == len(self.array):
20             self.__resize__()
21         self.array[self.rear] = new_item
22         self.rear = (self.rear+1) % len(self.array)
23         self.count+=1
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

- (a) Implement the method `__resize__(self)`, which is used by the `append` function. This method should double the size of the underlying array. It should also, if necessary, re-arrange the values of the instance variables.

## 程序代写代做 CS编程辅导

- (b) Implement the method `serve(self)`, which serves an item out of the Queue. This method never modifies the size of the underlying array, and raises an `Exception` if empty.



WeChat: cstutorcs

Assignment Project Exam Help

- (c) Implement the method `__str__(self)`, which returns a string representing the Queue, including all elements, separated by comma, from the front to the rear. For example, a Queue with two elements 1 and 2, at the front and the rear respectively, is represented by the string "[1,2]". An empty Queue will be "[]".

QQ: 749389476

<https://tutorcs.com>

- (d) In Big O notation what is the best and worst case for appending to a Queue with  $n$  elements and when do the cases occur. Explain by giving an example for each case.

Question 3 – Sorting Algorithms (9 marks = 3 + 3 + 3)

程序代写代做 CS编程辅导

Consider the BubbleSort, InsertionSort, SelectionSort, MergeSort, QuickSort and HeapSort algorithms we have seen in lectures, for sorting an array of length  $N$ . For those we have seen several versions (e.g. MergeSort and QuickSort) use the most efficient version we have seen.



- (a) Name those (if any) that are stable, and briefly explain why they are not stable.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

- (b) Name those (if any) that run in worst-case time  $O(N \log N)$ , and briefly explain how they manage to take  $O(N \log N)$ .

QQ: 749389476

<https://tutorcs.com>

- (c) Name those (if any) that run in best-case time  $O(N)$ , and briefly explain how they manage to take  $O(N)$ .

#### Question 4 – Hashing [8 marks]

程序代写代做 CS编程辅导

You have started coding a HashTable as follows:

```
1 class MyHashTable
2
3     def __init__(self, table_size):
4         self.table_size = table_size
5         self.array = [None] * table_size
6         self.count = 0
```



Assume you need to implement a function for your hash table. Each key to be hashed is a sequence of  $n$  integers, such as  $[10, 3, 5, 3, 20]$ . For example, a (key, value) pair to be stored could be  $([10, 3, 5, 3, 20], \text{"Introduction to CS"})$ . You are given the following three possibilities to choose from.

- (a) Returns the value of `random.randint(0, self.table_size-1)` (i.e., a random integer between 0 and the size of the table).
- (b) Returns the minimum value in the sequence to be hashed (3 in our example above) mod size of the table.
- (c) Returns the multiplication of all values in the sequence to be hashed ( $10 \times 2 \times 5 \times 3 \times 20$  in our example above) mod size of the table.

For each function explain briefly what are the disadvantages. Rank the three possibilities from best to worst.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutores@163.com

QQ: 749389476


<https://tutorcs.com>

Question 5 – Hash Tables [7 marks = 5 + 2]

程序代写代做 CS编程辅导

Consider a hash table implemented with an array of size 7.

- (a) Show in the final state of the array after inserting the numbers, 7, 3, 10, 5, 4, 11, the collisions are resolved using linear probing with the hash function provided in the table below.



7	0	0
3	3	1
10	3	2
5	5	3
4	4	4
11	4	5
		6

WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

- (b) What is the load of your hash table once all the numbers have been inserted?

<https://tutorcs.com>

# Question 6 – Linked Lists [9 marks = 6 + 3]

Consider the following linked List class, which you are in the process of defining:

```

1 class Node:
2
3     def __init__(self, data, link=None):
4         self.data = data
5         self.link = link
6
7 class List:
8
9     def __init__(self):
10        self.head = None
11
12    def is_empty(self):
13        return self.head is None

```



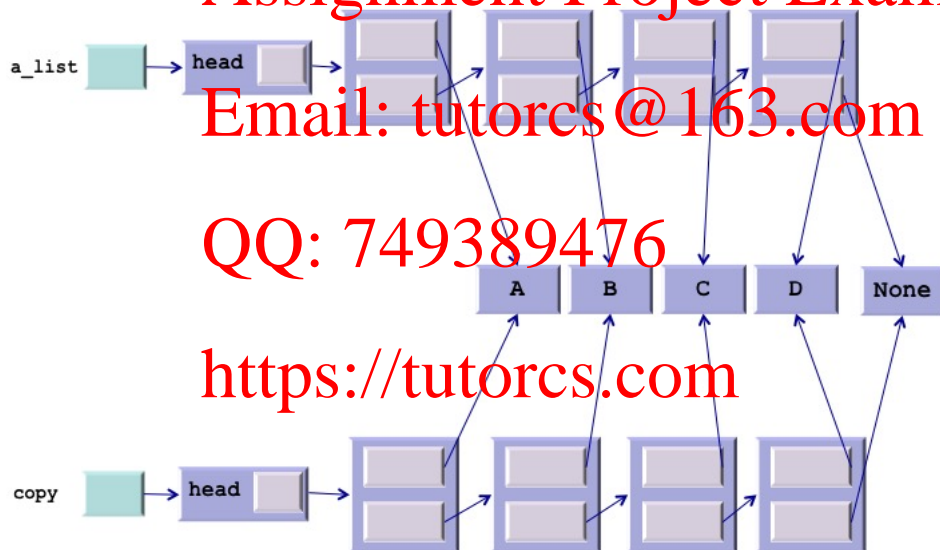
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutormcs@163.com

QQ: 749389476

https://tutorcs.com





## 程序代写代做CS编程辅导

- (a) Define a method `copy(self)` within the `List` class that returns a new linked list containing a copy of the nodes in `self`, in the given order and without modifying `self`. For example, given `a_list` with elements A,B,C, and D in the Figure above, the call to `a_list.copy()` returns the new list `copy` also shown in the figure, leaving `a_list` unchanged. If `a_list` is empty, `a_list.copy()` will return a new empty list. Do not assume the existence of other methods beyond those defined above.



WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

## 程序代写代做 CS编程辅导

(b) Below is a partial implementation of an Iterator for the `LinkedList` class defined in part a.

```
1 class MyLinkedListIterator:
2
3     def __init__(self, node):
4         self.current = node
```

Complete this implementation by writing down the methods `__next__(self)` and `__iter__(self)`.



WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

### Question 7 – Data Structures and Complexity [8 marks]

程序代写代做 CS编程辅导

Consider an **unsorted list** of  $N$  integers. Use Python to write a function that takes the list as a parameter and prints the two most frequent integers that appear more than once. For example, for a list `[6, 10, 6, 11]`, your function will print 6 and 4. Your implementation should have worst case time complexity  $O(n^2)$ , where  $n$  is the size of the list. You can assume and use any methods we have seen (`bubble_sort`, `insertion_sort`, `selection_sort`, `merge_sort`, `quick_sort`, `heap_sort`), and any data structure we have seen (`List`, `Stack`, `Queue`, `Tree`, `Heap`) – without writing them, including any methods we have defined. **Note:** Do not worry about exact method names or data structure names. As they are indicative, we are interested in your algorithmic reasoning, not syntax details.

WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

# Question 8 – Binary Trees [11 marks = 5 + 4 + 2]

Consider the partial implementation of BinarySearchTree class given below, which uses the TreeNode class:

```

1 class TreeNode:
2
3     def __init__(self, key, left=None, right=None):
4         self.key = key
5         self.left = left
6         self.right = right
7
8 class BinarySearchTree:
9
10    def __init__(self):
11        self.root = None
12
13    def is_empty(self):
14        return self.root is None
15
16    def LCA(self, key1, key2):
17        return self.LCA_aux(self.root, key1, key2)

```



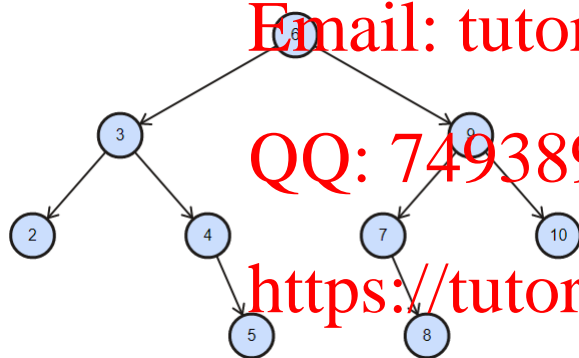
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



(Only keys depicted)

If the method LCA is applied to the binary search tree above, LCA(self, 2, 5) will return 3, LCA(self, 7, 4) will return 6, and LCA(self, 7, 8) will return 7.

## 程序代写代做 CS编程辅导

- (a) The Lowest Common Ancestor (LCA) of two nodes  $x$  and  $y$  in a binary tree is the node with the lowest key that has both  $x$  and  $y$  as descendants. Assuming  $\text{key1}$  and  $\text{key2}$  are both integers, implement the method `LCA_aux(self, current, key1, key2)` called from the `LCA` method of `Node` containing `key1` and `key2`. You can safely assume `key1` and `key2` do exist, in other words, the precondition of the method is that `key1` and `key2` are part of the Binary Search Tree. See the examples above for more details. A node is its own ancestor as shown by the last example.



WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

## 程序代写代做 CS编程辅导

- (b) Implement the method `traverse_preorder(self)` within the `BinarySearchTree` class above. The method should return a Python list containing all the keys in the tree, as given by a pre-order traversal. For example, for the BST in the figure above the method should return `[2, 4, 5, 9, 7, 8, 10]`. You can use the usual `append` method to add an element to the end of the Python list. For an empty heap, it should return an empty list.



WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

- (c) What is the best-case and worst-case time complexity of the most efficient implementation of `traverse_preorder(self)`? Explain. No explanation, no marks.

# Question 9 – Heaps [10 marks = 5 + 4]

Consider the partial implementation of MaxHeap class given below:

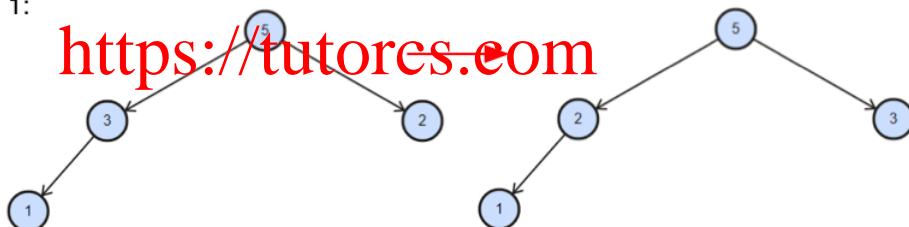
```

1 class MaxHeap:
2
3     def __init__(self, array):
4         self.array = array
5         self.count = len(array)
6
7     def swap(self, i, j):
8         self.array[i], self.array[j] = self.array[j], self.array[i]
9
10    def largest_child(self, k):
11        if 2 * k == self.count or self.array[2 * k][0] > self.array[2 * k + 1][0]:
12            return 2 * k
13        else:
14            return 2 * k + 1
15
16    def sink(self, k):
17        while 2 * k <= self.count:
18            child = self.largest_child(k)
19            if self.array[k][0] >= self.array[child][0]:
20                break
21            self.swap(child, k)
22            k = child

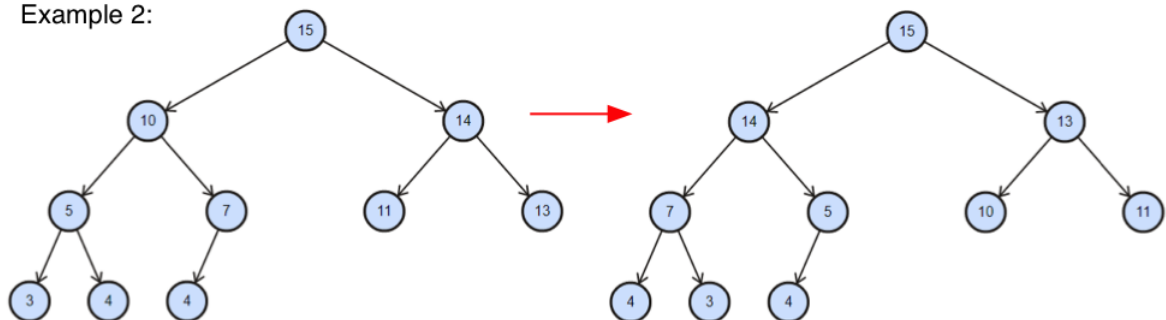
```

The underlying array stores tuples (key, value). The examples below depict keys only.

Example 1:



Example 2:



## 程序代写代做 CS编程辅导

- (a) Implement the method `swap_children(self)`, which modifies the heap structure to swap the immediate children of every node (if any), sinking as necessary to keep the structure a valid max heap. For example, in the figures above, if the method is applied to the left, it will modify it to be the one on the right.



WeChat: cstutorcs

Assignment Project Exam Help

Email: [tutorcs@163.com](mailto:tutorcs@163.com)

QQ: 749389476

<https://tutorcs.com>

- (b) Draw the array behind a heap (using the convention seen in the lectures) after inserting the keys 15, 32, 17, 51, 29, 10, 23 into the max heap and then deleting 51 and 32 (no need to depict values, only keys).



## 程序代写代做 CS编程辅导

### Question 10 – Classes, Objects, and Namespaces [8 marks]

Examine the following Python code:

```
1 class Car:
2
3     numberOfTyres = 4
4     steeringLocation = "Left"
5     engineLocation = "Front"
6     regionCode = "UK"
7
8     def __init__(self, brand, enginePower):
9         self.brand = brand
10        self.enginePower = enginePower
11
12    def setSeatNumber(self, numberOfSeat):
13        self.seatNumber = numberOfSeat
14
15    def setColour(self, colour):
16        Car.colour = colour
17
18    car2 = Car("Nissan", 2400)
19    car2.setColour("Green")
20    car2.engineLocation = "Back"
21    car2.numberOfTyres = 6
22    car1 = Car("Toyota", 1000)
23    car1.setSeatNumber(3)
24    car1.steeringLocation = "Right"
25    car1.colour = "Red"
26    car1.regionCode = 1000000
27    print(car2.colour) #1
28    print(car1.colour) #2
29    print(car2.steeringLocation) #3
30    print(car1.engineLocation) #4
31    Car.numberOfTyres = 3
32    car1.enginePower = 500
33    print(car2.numberOfTyres) #5
34    print(car1.numberOfTyres) #6
35    print(car2.enginePower) #7
36    print(car1.seatNumber) #8
```

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

- (a) Provide the result of each print statement (marked with comments from #1 to #8) – **next to the comment above**. If the results is an error, explain why assuming the execution will continue after executing the Python code above.

END OF EXAM.