

Name and Username (abc123): _____

- You have 1 hour and 30 minutes to complete this exam.
- It contains **10** problems, some with multiple parts.
- There are 4 pages to the exam.
- The exam has a total of 140 points.
- Select questions totaling 100 points to complete.
- If you submit more than 100 point worth of questions, the **lowest** scoring questions will be counted towards your grade.
- Submit your answers as a text file. [username]_midterm.txt
- This exam is limited **open book**. You may use:
 - Zybooks
 - Any material posted in BBLearn (lectures, homeworks, etc)
 - Any material you created (Homework answers, notes, etc)
 - The website <https://www.wolframalpha.com> for calculations
 - The Python 3 Docs <https://docs.python.org/3/index.html>
 - A calculator (physical or software)
 - Python 3
 - Any IDE or text editor of your choosing.
- You may use scrap paper.
- You may not talk to/message anyone during the exam. All work must be your own.
- You may not search websites not listed above.
- If you are unsure of a instruction, write how you interpreted it before your answer.
- **Good Luck!**

Question	Points	Score
1	10	
2	10	
3	10	
4	20	
5	20	
6	20	
7	10	
8	20	
9	10	
10	10	
Total:	140	

1 Essay Questions

These questions are essays. They should be answered in plain text. You might have to make tables or write equations in plain text.

Question 1 : 10 points

Sort the following from *slowest growth rate* to *fastest growth rate*. Let $f(x)$ grow slower than $g(x)$ if and only if $f(x) = o(g(x))$. Clearly label the fastest and slowest growth rate functions.

$$\begin{array}{ccc} 2^{\log_2(n)} & \log_2(n) & n^2 \\ n! & e^n & 12 \\ n \log_2(n) & n^5/n^2 & \frac{1}{n} \\ n * \sqrt{n} & & \end{array}$$

Question 2 : 10 points

Insert the following list of numbers into both a closed and open hash table. In the comments of your file, show what both hash tables would look like if the following numbers were inserted in order.

Insert: 69, 53, 90, 82, 50, 99, 62, 87, 26, 79

Use the hash function define below. For the closed table, use linear collision resolution. Attempt spaces in the order $(h(i) + k) \bmod 10$ for $k = 0 \dots 9$ to find the next open space.

$$h(i) = i \bmod 10$$

Array indexes should be labeled 0-9 for both parts.

- (a) (5 points) Show the **Closed Hash Table** with 10 spaces using the given values.
- (b) (5 points) Show the **Open Hash Table** with 10 spaces using the given values.

Question 3 : 10 points

Compare a linked list and an Array.

Give 2 situations where an array would be better to use than a linked list.

Give 2 situations where a linked list would be better to use than an array.

Give 1 situation where it won't matter which was used.

2 points per situation described.

2 Pseudocode Questions

All the following questions are answered in pseudocode. That means your code should appear reasonably similar to working code, but does not need to actually run.

Question 4 : 20 points

You work at a company that makes specialized security briefcases. You need to test a new model to see how many floors up it can be dropped from and not smash. Assume the test is boolean (Yes/No). Either the case breaks enough someone can get the contents or they cannot access the contents.

Given a building with N floors, come up with an Algorithm to find the lowest floor the briefcase smashes from. Your goal is to optimize the number of tests. Come up with an algorithm that throws as few briefcases as possible. A briefcase can only be thrown once, even if it does not break.

Give a pseudocode function `findFloor(numFloors)`. Assume you have a function `testFloor(f)` that return True if the briefcase breaks when thrown from floor f and False otherwise.

Question 5 : 20 points

Nuts and Bolts by G. J. E. Rawlins

You have a mixed pile of N nuts and N bolts and need to quickly find the corresponding pairs of nuts and bolts. Each nut matches exactly one bolt, and each bolt matches exactly one nut. By fitting a nut and bolt together, you can see which is bigger, but it is not possible to directly compare two nuts or two bolts.

Give a pseudocode function for solving the problem. You **may not** compare nuts to each other. You **may not** compare bolts to each other. Come up with the most efficient solution you can.

The function `sortNB(Nuts,Bolts)` should take two lists and sort them in order such that `Nuts[0]` matches `Bolts[0]` and they are both the smallest.

Assume you have a function `compare(Nut[i],Bolt[j])` that returns

1. -1 if the bolt is too small to fit in the nut
2. 0 if the bolt matches the nut
3. 1 if the bolt is too big to fit in the nut

Question 6 : 20 points

Assume you have a stack and a queue. The stack starts with many values, the queue starts empty. Use the queue to reverse the stack. Write a pseudocode function `reverseStack(S,Q)` that uses queue Q to reverse all the values in stack S . The function has no return value, but when it completes the Stack S is reversed.

The queue Q only has methods: `enqueue`, `dequeue`, and `isEmpty`

The stack S only has methods: `push`, `pop`, and `isEmpty`

3 Coding Questions

These questions must be written in working Python 3 code.

Question 7 : 10 points

Assume you have a `Node` class as shown below. Write a python3 function `reverse(head)` that takes the first node in a linked list and reverses the list. The function returns the new head of the list. You **may not** create any new nodes. Just change the values/next points of existing nodes.

```

1 class Node:
2     def __init__(self, v, n):
3         self.value=v
4         self.next=n
5     def __str__(self):
6         return str(self.value)+"->" + str(self.next)
7 #Example: Your solution should work on the general case
8 head=Node(30,Node(41,Node(25,Node(10,None))))
9 print(head) #30->25->41->10->None
10 head = reverse(head)
11 print(head) #10->41->25->30->None

```

Question 8 : 20 points

We can express insertion sort as a recursive procedure. In order to sort $A[1 \cdots n]$, we recursively sort $A[1 \cdots n-1]$ then insert $A[n]$ into the sorted array $A[1 \cdots n-1]$. In Python 3, implement a recursive version of insertion sort.

Question 9 : 10 points

Write 10 pytest tests the show the below function works as intended for $a \geq 0, b > 0$. Each test should assert a specific answer is correct.

```

1 def remainder(a,b):
2     if a < b:
3         return a
4     else:
5         return remainder(a-b,b)

```

Question 10 : 10 points

Create a function `timeZeta()` that uses `timeit` to time the following function for inputs $a = 3, 9, 27, 81, \dots, 2187$. ($3^7 = 2187$). Print the results in two column. First column is the value of a . Second column is the time. Set number to be 1000.

```

1 def zeta(a):
2     total=0
3     for i in range(1,100):
4         total=total+(1/(i**a))
5     return total

```