Computer Science and Information Management

School of Engineering and Technology

AT70.02 Algorithm Analysis and Design

August 2022 Semester

MIDTERM EXAM

Instructor: CHAKLAM SILPASUWANCHAI
Time: 3 hours

STUDENT NAME: _________________________ STUDENT ID. NO. __________________

- This exam accounts for **30%** of the overall course assessment**.**
- This exam is **open-booked; open-internet.**
- **Don't write only the answer. "Precise" proof (and steps) are required for ALL questions.**
- The completed exams shall be submitted to the Google Classroom

EXAMINATION RULES:

- For **offline students**, you may leave the room temporarily with the approval and supervision of the proctors. No extra time will be added to the exam in such cases.
- For **online students**, you are required to turn on your webcam during the entire period of the exam time
- Students will be allowed to leave at the **earliest 45 minutes** after the exam has started
- All work should belong to you. A student should <u>NOT</u> engage in the following activities which proctors reserve the right to interpret any of such act as a**cademic dishonesty without questioning**:
    - Chatting with any human beings physically or via online methods
    - Plagiarism of any sort, i.e., copying from internet sources or friends
- No make-up exams are allowed. Special considerations may be given upon a valid reason for unpredictable events such as accidents or serious sickness.

1. Illustrate QUICKSORT on array *A* = <6, 8, -2, 1, 7, 6, 4, 3>. Use Median of Three (*1pt.)*
2. Illustrate COUNTING SORT on array *A* = <1, 0, 4, 4, 1, 3, 6, 3, 7>. (*1pt.)*
3. Illustrate HEAPSORT of [6, 3, 9, 5, 7, 8]. (*1pt.)*
4. Find the MAXIMUM SUBARRAY of A = <-7, 5, -6, 5, -2, 6> (*1pt.)*
5. Insert the following A = <3, -2, 9, 4, 5, 1,4, 3, 2> into the BINARY TREE.(*1pt.)*
6. Continue above, delete 9. (*1pt.)*
7. Illustrate the DOUBLE HASHING where h1(k) = k mod m, h2(k) = 1 + (k mod (m-1)). Let k = {15, 63, 20, 21, 32}, m = 7. Perform simply linear probing on h1 in the case of collision. (*1pt.)*
8. Solve the followings with Master Theorem: *(1pt each)*
    a. $T(n) = 16\, T(n/2) + n^3$
    b. $T(n) = 4\, T(n/2) + 2^{\log n}$
    c. $T(n) = 3\, T(n/2) + n \log n$
9. Explain the space-performance tradeoff in hashing. *(1pt.)*
10. Why are we required to check $a\, f\left(\frac{n}{b}\right) \leq cf(n)$ for case 3 in the Master theorem? *(1pt.)*
11. Compare these two functions. Which one is faster? Argue with asymptotic and recurrence analysis. For recurrence, you can use substitution, tree, or Master theorem. (*1pt.)*

```python
def reverse_recursive(s):
    if s == "":
        return s
    else:
        return reverse_recursive(s[1:]) + s[0]

def reverse_iterative(s):
    s1 = ''
    for c in s:
        s1 = c + s1  # appending chars in reverse order
    return s1

print(reverse_recursive("chaky"))  #ykahc
print(reverse_iterative("chaky"))  #ykahc
```

Coding

12. Given three lists - A, B, and C, write a python function `check_null()` such that it returns `True` if there is NO element x such that $x \in A,\ x \in B,\ x \in C$. Otherwise, returns `False`.
    a. Write an algorithm of complexity running time is $O(n^3)$. Don't forget to provide a simple proof that the complexity is really $O(n^3)$ (*2pts.)*
    b. Improve the algorithm to $O(n^2)$. Provide the proof as well. (*2pts.)*

GOOD LUCK!