四川大学期末考试试题（闭卷）

（2021——2022 学年第 1 学期）　　A 卷

课程号：304133030　　　　课序号：01　　　课程名称：数据结构与算法分析（全英文）

任课教师：徐文政　　　　成绩：

适用专业年级：计算机学院 2020 级试验班　　学生人数：15　　印题份数：16

学号：＿＿＿＿＿＿＿＿＿　　姓名：＿＿＿＿＿＿＿＿＿

考 生 承 诺

　　我已认真阅读并知晓《四川大学考场规则》和《四川大学本科学生考试违纪作弊处分规定（修订）》，郑重承诺：

1、已按要求将考试禁止携带的文具用品或与考试有关的物品放置在指定地点；

2、不带手机进入考场；

3、考试期间遵守以上两项规定，若有违规行为，同意按照有关条款接受处理。

考生签名：＿＿＿＿＿＿＿＿＿

Instructions to Candidates

This is a closed-book examination.

There are SEVEN problems in this examination paper.

The total marks of this examination paper are 100.

The mark of each question is indicated at the end of the question.

Write down your answers on the answer sheets provided.

Time Allowed:　**2 hours.**

Approved calculators may be used (Get permission from the invigilator).

1.　Fill blanks　(2 points *5=10　points）

Given a complete K-ary tree (e.g., $K=3$) with height h ($h>0$), there are (　　　　　) nodes at the i-th level with $0 \leq i \leq h-1$, and there are at least (　　　　　) node(s)，　at most (　　　　　) nodes at the bottom level (i.e., the h-th level). Also, there are at least (　　　　　) nodes, at most (　　　　　) nodes in the tree.

2. Program　（15 points）

Given a linked list A with n nodes, split it into two sublists B and C, such that the numbers of nodes in sublists B and C are almost balanced, and there are $\lfloor n/2 \rfloor$ nodes in sublist B and $\lceil n/2 \rceil$ nodes in sublist C. The declaration of the function is as follows, where hA points to the head node of list A, while hB and hC point to the head nodes of sublists B and C, respectively, and n is the number of nodes in list A. Initially, both sublists B and C consist of only a head node.

void ListBalancedSplit(Node *hA, int n, Node *hB, Node *hC);

Requirement: Please first describe your basic idea behind. Then, write down your algorithm.

3. Please answer the following question　（15 points）

Given the structure of a max heap with 9 keys with each node in the heap having no more than 3 children. Usually an array is used to store a heap. Given an array *A* with its size 9. Please show how does the array store the heap? Assume that a key is stored at array *A[i],* where are its *three children* stored in the array, respectively? Also, where is its *parent* stored in the array? Assume that the root is stored at *A[0].*
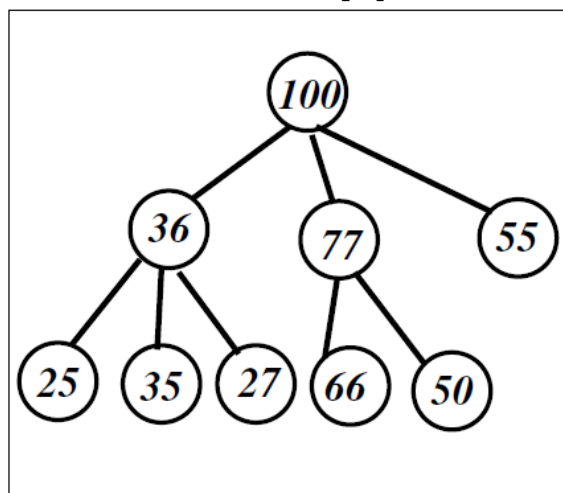
4. Please answer the following question （15 points）

Given a *K*-ary tree (e.g., *K*=3) with the number of children of each node being no greater than *K*, assume that there are *n* internal nodes in the tree with *n*≥1. What is the minimum number of leaves in the tree and what is the tree topology? What is the maximum number of leaves in the tree and what is the tree topology? Also, please prove the maximum number of leaves?

5. Program （15 points）

Given an ***undirected*** graph *G=(V, E)*, graph *G* is stored in a computer by an adjacency List. Please devise an algorithm to detect whether there is a cycle in *G*? Also, please show the time complexity of your proposed algorithm, and let *n=|V|* and *m=|E|*.

Requirement: Please first describe the basic idea behind. Then, write down your algorithm.

6. Program (15 points）

Given an undirected, weighted graph *G=(V, E)*, let *n=|V|* and *m=|E|*, where indices of the nodes in *G* are between 0 and *n-1*. Graph *G* is stored in a computer by an adjacency List. Find the shortest paths from a given source node *s* to other nodes by applying the Dijkstra' algorithm, where 0≤*s*≤ *n-1*. Assume that a minimum heap with its data type *MIN-HEAP* is adopted, and each node in the heap consists of: (i) node id; and (ii) the key of the node, e.g., the key means the estimated shortest distance from the source *s* to the node. The minimum heap supports following four operations: (i) build an empty heap, i.e., function *void build ()*; (ii) insert a node id to the heap, i.e., function *void insert(int id, double key)*; (iii) remove the node with the minimum key from the heap and return the id of the node, i.e., function *int removeMin()*; and (iv) decrease the key of a node id, i.e., function, *void decreaseKey(int id, double newKey)*. Please show the time complexities of the last three operations of the heap, respectively. Also, write down the Dijkstra' algorithm by adopting the minimum heap, and analyze the algorithm time complexity. Notice that you can directly invoke the four functions *build*, *insert*, *removeMin*, and *decreaseKey* in your algorithm. For example, see the two lines of codes:

    MIN-HEAP minHeap; // declare a min heap

    minHeap.build(); // build an empty heap

*Requirement***:** Please first describe the basic idea behind. Then, write down your algorithm.

7. Please answer the following question （15 points）

Consider the quickSort algorithm. Assume that 31 is the pivot. Please show the process of partitioning the numbers in the following lists into three parts: (1) the numbers in the left part are strictly less than 31; (2) the pivot 31 itself; and (3) the numbers in the right part are no less than 31.

| 35 | 33 | 42 | 10 | 14 | 19 | 27 | 44 | 26 | 31 |

– End –