Prerequisite Questions

Tunghwa Wang, Ph.D.
COEN 279 Computer Algorithm
Department of Computer Engineering
Santa Clara University

Name and Education Background

(_____)

( ) 1. Computer science related
( ) 2. Engineering related
( ) 3. Mathematics related
( ) 4. Others (_____)

Knowledge in Programming Language

One of the following listed programming languages is required:
( ) 1. C
( ) 2. C++
( ) 3. Java
( ) 4. Python

Knowledge in Mathematics

( ) 1. Calculus
( ) 2. Number theory
( ) 3. Applied math
( ) 4. Others

Knowledge in Data Structures

( ) 1. Array
( ) 2. Stack
( ) 3. Queue
( ) 4. Hash table
( ) 5. Linked list
( ) 6. Heap
( ) 7. Tree
( ) 8. Graph

Review Questions

1. [doubly linked list] If defined as followings:

```
typedef struct dll_s
{   struct dll_s *pNext;
    struct dll_s *pPrev;
    int val;
} dll_t;
dll_t *my_list = NULL;
```

Please implement the following functions:

1] void insert(int val);
to insert val into my_dll in ascending order.

2] void delete(int val);
to delete val from my_dll.

3] What is the advantage of implementing as
void insert(dll_t *list, int val);
void delete(dll_t *list, int val);

4] What is the advantage of implementation as
int insert(dll_t *list, int val);
int delete(dll_t *list, int val);

2. [tree] If defined as followings:

```
typedef struct tree_s
{    tree_s *pLeft;
     tree_s *pRight;
     int val;
} tree_t;
tree_t *my_bst = NULL;
```

Please implement the following functions:

1] int insert(tree *bst, int val);
to insert val into the binary search tree my_bst.

2] int delete(tree *bst, int val);
to delete val from the binary search tree my_bst.

3] void print(tree *bst);
to print all values of the tree nodes in ascending order.

4] void print(tree *bst);
to print all values of the tree nodes in descending order.

3. [hash table] What is the advantage of choosing the 2^N as size of hash table?

4. [prime numbers] Implement sieve algorithm to print all prime numbers less than 100.

5. [Fibonacci numbers] Implement a function to print the first 10 Fibonacci numbers defined as F(n) = F(n-1) + F(n-2) where F(0)=1 and F(1)=1.