CSCI 2520 Data Structures and Applications

Assignment One

Deadline: 23:55, Mar. 16, 2020

Total Marks: 100

Submission:

In this Assignment, you need to answer question 1&2 in one pdf file. For question 3&4, you need to provide .c file for each question, which can be compiled and give correct answer, the name should be q3.c and q4.c respectively.

Then compress all 3 files (one pdf file for q1&q2 and two .c files for q3&q4 respectively) as one zip named as `your_student_id_assign1.zip` and submit it via blackboard.

1. Stack (20 marks)

Given two sequences:

Sequence A: 4 3 1 2 6 5

Sequence B: 1 2 3 4 5 6

You need to write down how to use STACK to get the Sequence B from Sequence A.(Write the steps **ONLY** using the **Pop** and **Push** functions)

void Push(stackADT stack, int element);

stackElementT Pop(stackADT stack);

S = EmptyStack();

For example, if you need to **push 4** into the stack, just write down **Push(S, 4)**; if you need to **pop** one element from the stack, just write down **Pop(S)**.

2. Hashing (20 marks)

(a) Data entries with the following keys are inserted into or deleted from a hashtable of size 8 using (i) linear probing and (ii) quadratic probing, respectively. The hashtable is empty initially and hash code begins from 0. Draw the final hash tables. You may assume that lazy deletion is implemented, i.e., when a data entry is deleted, it is replaced by an 'X'. (16 marks)

Action	Order of input data	Hash Code
Insert	Stack	1
Insert	Queue	2
Insert	List	3
Insert	Table	1
Delete	Stack	1
Insert	Tree	2
Insert	Heap	7
Delete	List	3
Insert	Graph	6

(b) Is there anything problem in part (a)? What is it? How can the issue be avoided? (4 marks)

3. Stack Programming (30 marks)

Given a string S containing the characters '(', ')' and 'T'. All the parentheses are in pairs.

1. Parentheses can be nested, like '(())' and '(()())',
2. There is exactly one 'T' in a given string S.

Write a program using the stack ADT to find out the number of pairs of matching parentheses that contain 'T'.

Examples:

input: `(((T)))(())`

output: `3`

input: `(()(T))()`

output: `2`

4. Reverse Linked List (30 marks)

Write a function to reverse a singly linked list.

Format of functions:

```
List Reverse( List L );
```

where List is defined as the following:

```
typedef struct Node *PtrToNode;
typedef Ptr Node List;
typedef PtrToNode Position;
struct Node {
    ElementType Element;
    Position Next;
};
```

The function **`Reverse`** is supposed to return the reverse linked list of **`L`**, with a dummy header.

Sample Input:

```
5
1 3 4 5 2
```

Sample Output:

```
1 3 4 5 2
2 5 4 3 1
```

Sample program is provided in **assign1-4.c**, you just need to append the implementation of function **`Reverse`** to the tail of the program.