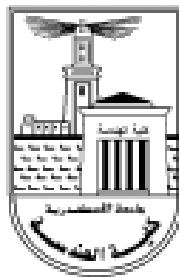


Data Structures CE233

Alexandria University

Faculty of Engineering
SSP- Program.
Summer Course



Assignment (3)

Assigned: 4/8/2014

Due: 10/8/2014

Prof. Dr. Magdy A. Ahmed

- 1) The polynomial $A(x) = a_{m-1}x^{e_{m-1}} + \dots + a_0x^{e_0}$ where a_i are nonzero coefficients and e_i are nonnegative integer exponents such that $e_{m-1} > e_{m-2} > \dots > e_1 > e_0 \geq 0$. We can represent the polynomials as linked lists such that each term will be represented as a node, using the following structure :

Coef	Expon	Link
------	-------	------

- (a) Let A and B be pointers to two polynomials. Write an algorithm to compute the sum polynomial $D = A + B$ using a singly linked list.
- (b) Rewrite Exercise (a) using a circular linked list.
- (c) Let A be a pointer to a polynomial. Write an algorithm to evaluate the polynomial A at point x, where x is some integer number.
- (d) Rewrite Exercise (c) using a circular linked list.
- (e) Let A and B be pointers to two polynomials. Write an algorithm to compute the product polynomial $D = A * B$ using a singly linked list.
- (f) Rewrite Exercise (e) using a circular linked list.

Programming Assignment # 3

Write a C Program(s) to implement the polynomial algorithms described in the above problem.

- Use the insertion algorithm to create polynomial A and polynomial B.
- Use the display algorithm to print out the polynomial D.
- Test your program using the following polynomials :

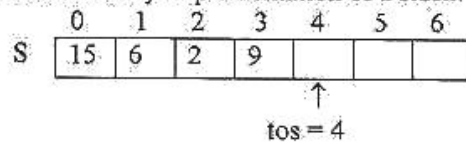
$$A = 3x^{14} + 2x^8 + 1$$

and

$$B = 8x^{14} - 3x^{10} + 10x^6$$

Data Structures CE233

1- Given an array implementation of a stack:



- How many elements are in the stack?
 - Which is the top element?
 - Draw the stack after the calls PUSH(S,17); PUSH(S,3)
 - What is tos now?
 - What is returned by the call POP(S)?
 - Draw the stack after the call POP(S)
 - What is the top element now?
- 2- What are a stack underflow and a stack overflow?
- 3- Show the corrections of the following procedures indicating what they do

STACK-EMPTY(S)

```
Begin
  if tos = 0
    then
      return TRUE
    else
      return FALSE
  endif
End.
```

PUSH(S, x)

```
Begin
  tos ← tos + 1
  S[tos] ← x
End.
```

POP(S)

```
Begin
  if STACK-EMPTY(S)
    then
      error "underflow"
    else
      tos ← tos - 1
    endif
  return S[tos + 1]
End
```

- 4- Illustrate the result of each of the operations PUSH(S, 4), PUSH(S, 1), PUSH(S,3), POP(S), PUSH(S,8), POP(S) on an initially empty stack S stored in array S [0..5].

Programming Assignment # 4

Write a complete C program that transforms an infix expression into its postfix equivalent, and then evaluates the resulting postfix expression.

Data Structures CE233

More Exercises: (Solved)

1- For the following 2-D array, how will it be represented in memory using row major and column major order?

5	4	3	0
0	6	2	1
4	3	7	8

How will we calculate the address of $A[2][1]$ (in the 2 cases row and column major orders)?

2- For stack (As Array Representation) Implement the structure of it and all functions needs (initialization, push, pop, isFull, and isEmpty), test your Code then do some pushes then do the following:

- Get Sum and Count of all elements in stack.
- Reorder the stack where all even in up and all odd in down.
- Reverse stack order in place.
- Sort stack.
- Insert in sorted stack.
- Merge two ordered stack into one ordered stack.
- Delete element from stack at k_{th} position (k_{th} from top).
- Check if the sum of first half equals to the sum of the second half of stack

3- Write the following expressions $(4+5)*(4-2) + 7*3$ in postfix notation and use stack to evaluate the expression and get its value showing the content of stack in each step and sequence of operations (push, pop).

After you write the previous expression in postfix notation check it with this,
 $4\ 5\ +\ 4\ 2\ -\ *\ 7\ 3\ *$

What is the missing operation to correct the above expression?

4- Use Stack to show that this expression is either valid or invalid $4\ 5\ 3\ *\ +\ 4\ 5\ *\ +\ -$, if it is valid show its value and if it is not show one option to correct it.

5- A palindrome is a word that is the same when spelled from the front or the back. For example “reviver” is palindrome. Write an algorithm that uses a stack to determine if a word is a palindrome.

6- Write the prefix form of the following expressions :

- $a/b-c+d*e-a*c$
- $a*(b+c)/d-g$
- $(a+b)*d+e/(f+a*d)+c$