**CSC248 – Fundamentals of Data Structure**

**Academic Session October 2023 – February 2024**

**Lab Assignment 5 – Stack (BI)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Outcomes (CO)** | **LO1** | **LO2** | **LO3** |
| CO1 |  |  |  |
| CO2 | √ | √ | √ |
| CO3 |  |  |  |

1. Write a program to convert the decimal number into hexadecimal number by using stack concept. –(**NOTE:** use built-in ArrayList method ONLY)

**Example:**

145210 = 5AC16

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner intInput = new Scanner(System.in);

        System.out.println("Welcome to the Decimal to Hexadecimal Converter!\n");

        ArrayList<Integer> numbers = new ArrayList<Integer>();

        // convert decimal to hexadecimals

        // use stack concept but with arraylist

        System.out.print("Enter a decimal number: ");

        int decimal = intInput.nextInt();

        // last in first out

        while (decimal > 0) {

            int remainder = decimal % 16;

            numbers.add(remainder);

            decimal /= 16;

        }

        System.out.print("\nHexadecimal: ");

        for (int i = numbers.size() - 1; i >= 0; i--) {

            if (numbers.get(i) > 9) {

                // the number 55 is added to the retrieved number. The reason for adding 55 is

                // because in the ASCII table, the character ‘A’ (which represents 10 in

                // hexadecimal) is at position 65. So, if you have a number like 10, adding 55

                // will give you the ASCII value of ‘A’.

                char hex = (char) (numbers.get(i) + 55);

                System.out.print(hex);

            } else {

                System.out.print(numbers.get(i));

            }

        }

        System.out.println();

        System.out.println("\nThank you for using the Decimal to Hexadecimal Converter!");

        intInput.close();

    }

}

Sample Output

A black screen with white text

Description automatically generated

1. Write a program to evaluate a postfix arithmetic expression. You **MUST** use User Defined Type(UDT) Linked List

**Example:** 2 6 \* 3 - / 5 5 \* +

**Result :** 28

import java.util.Scanner;

class Node {

    int data;

    Node next;

}

class LinkedLists {

    Node top;

    LinkedLists() {

        top = null;

    }

    void push(int data) {

        Node node = new Node();

        node.data = data;

        node.next = top;

        top = node;

    }

    int pop() {

        if (top == null) {

            System.out.println("Stack Underflow");

            return -1;

        } else {

            int temp = top.data;

            top = top.next;

            return temp;

        }

    }

}

public class Main {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the postfix expression: ");

        String exp = input.nextLine();

        LinkedLists stack = new LinkedLists();

        String[] tokens = exp.split(" ");

        for (int i = 0; i < tokens.length; i++) {

            if (tokens[i].equals("+")) {

                int a = stack.pop();

                int b = stack.pop();

                stack.push(a + b);

            } else if (tokens[i].equals("-")) {

                int a = stack.pop();

                int b = stack.pop();

                stack.push(b - a);

            } else if (tokens[i].equals("\*")) {

                int a = stack.pop();

                int b = stack.pop();

                stack.push(a \* b);

            } else if (tokens[i].equals("/")) {

                int a = stack.pop();

                int b = stack.pop();

                stack.push(b / a);

            } else {

                stack.push(Integer.parseInt(tokens[i]));

            }

        }

        System.out.println("postfix evaluation: " + stack.pop());

    }

}

Sample Output

