

1. Stack-г хэрэгжүүл Хэрэгжүүлэгч функц(`stk_push`, `stk_pop`, `stk_Read`)

```

import java.util.Scanner;
import java.util.Stack;

public class Lab_Stack {
    public static void main(String[] args) {
        Stack<Integer> stk = new Stack<Integer>();
        Scanner sc = new Scanner(System.in);
        int value = 0, type = 0, number;
        while (value != 0) {
            System.out.println("1. Тоо нэмэх 2. Тоо хасах 3. Унших 4.
Гаргах");
            type = sc.nextInt();
            switch (type) {
                case 1: {
                    stk_push(stk);
                } break;
                case 2: {
                    stk_pop(stk);
                } break;
                case 3: {
                    stk_Read(stk);
                } break;
                case 4: {
                    System.out.println("Гарлаа !!!");
                    System.exit(0);
                } break;
            }
        }
    }

    private static Stack stk_push(Stack<Integer> stack) {
        int val;
        Scanner sc = new Scanner(System.in);
        System.out.print("Өгөгдөл: ");
        val = sc.nextInt();
        stack.push(val);
        return stack;
    }

    private static Stack stk_pop(Stack<Integer> stack) {
        if (!stack.empty()) {
            stack.pop();
            System.out.println("Хасагдлаа");
        }
        else System.out.println("Хоосон байна?");
        return stack;
    }

    private static void stk_Read(Stack<Integer> stack) {
        System.out.println(stack);
    }
}

```

## Output:

```

1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах
1
Өгөгдөл: 3
1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах
1
Өгөгдөл: 2
1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах
3
[3, 2]
1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах
2
Хасагдлаа
1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах
3
[3]
1. Тоо нэмэх 2. Тоо хасах 3. Унших 4. Гарах

```

2. Өгсөн тектэнд хамгийн их давтагдан орсон 2 үгийг гарга  
 HashMap ашигласан key-р нь үг дамжуулж value-тоолсон утгаа дамжуулна  
 Яагаад 4 давталт ашигласан вэ?  
 1) 2 давталт нь хамгийн их давтагдсан 2 үгийг эрэмбэлж хайна  
 2) 2 давталт нь хамгийн их дэвтэгдсэн үгийн мэдээллийг хэрэглэгчид харуулах

```

import java.io.FileNotFoundException;
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class Max_Count_Text_2 {
    static void CountWords(String filename, Map<String, Integer> words)
    throws FileNotFoundException {
        Scanner file = new Scanner(filename);
        while (file.hasNext()) {
            String word = file.next();
            Integer count = words.get(word);
            if (count != null)
                count++;
            else
                count = 1;
            words.put(word, count);
        }
    }
}

```

```

    }
    file.close();
}

public static void main(String[] args) throws FileNotFoundException {
    Map<String, Integer> words = new HashMap<String, Integer>();
    Scanner sc = new Scanner(System.in);
    System.out.println("Үгээ оруулах: ");
    String text = sc.nextLine();
    int sameSize = 0;

    String[] textArr = text.split(" ");
    CountWords(text, words);
    int max = 0, nextmax = 0;
    for (Map.Entry<String, Integer> entry : words.entrySet()) {
        if (entry.getValue() > max) {
            max = entry.getValue();
        }
    }

    for (Map.Entry<String, Integer> entry : words.entrySet()) {
        if (max == entry.getValue()) {
            System.out.println(entry.getKey() + "=>" + entry.getValue());
            sameSize++;
            entry.setValue(0);
        }
    }

    if(sameSize<2)
    {
        for (Map.Entry<String, Integer> entry : words.entrySet()) {
            if (entry.getValue() > nextmax) {
                nextmax = entry.getValue();
            }
        }

        for (Map.Entry<String, Integer> entry : words.entrySet()) {
            if (nextmax == entry.getValue()) {
                System.out.println(entry.getKey() + "=>" +
entry.getValue());
            }
        }
    }
}
}

```