

BEN-GURION UNIVERSITY OF THE NEGEV

DATA STRUCTURES

202.1.1031

---

## Assignment No. 4 - Solution

---

*Authors:*

**Name 1** (I.D 1)

**Name 2** (I.D 2)

Publish date: 20.5.2023

Submission date: 10.06.2023

אוניברסיטת בן-גוריון בנגב  
Ben-Gurion University of the Negev



# Contents

<b>1</b>	<b>An example of Pseudo-Code</b>	<b>3</b>
<b>2</b>	<b>Skip-List</b>	<b>3</b>
2.1	Warm-Up and Familiarization . . . . .	3
2.1.1	Implementation of Abstract Functions . . . . .	3
2.1.2	Analysis of the Probabilistic Process . . . . .	3
2.1.3	Analysis of the operations . . . . .	3
2.2	Order Statistics . . . . .	4
<b>3</b>	<b>Hashing</b>	<b>5</b>
3.1	Introduction . . . . .	5
3.1.1	Hash Functions . . . . .	5
3.2	Hash Implementations . . . . .	5
3.3	Hash Tables . . . . .	5
3.3.1	Introduction . . . . .	5
3.4	Theoretical Questions . . . . .	6
<b>4</b>	<b>Designing a data structure according to given specifications</b>	<b>7</b>

## Tasks and Questions

2.1	Answer	...	3
2.2	Answer	...	3
2.3	Answer	...	3
2.4	Answer	...	3
2.5	Answer	...	3
2.6	Answer	...	3
2.7	Answer	...	4
2.8	Answer	...	4
2.9	Answer	...	4
2.10	Answer	...	4
2.11	Answer	...	4
2.12	Answer	...	4
3.1	Answer	...	5
3.2	Answer	...	5
3.3	Answer	...	5
3.4	Answer	...	5
3.5	Answer	...	5
3.6	Answer	...	5
3.7	Answer	...	5
3.8	Answer	...	5
3.9	Answer	...	5
3.10	Answer	...	5
3.11	Answer	...	5
3.12	Answer	...	6
3.13	Answer	...	6
3.14	Answer	...	6
3.15	Answer	...	6
3.16	Answer	...	6
4.1	Answer	...	7

# 1 An example of Pseudo-Code

Remove this example:

---

**Function:** FindMax( $L$ )

---

**Input:** A List  $L$  of  $n$  *comparable* values.

**Output:** The maximal value within  $L$

```
1:  $max \leftarrow L[0]$ 
2: for  $1 \leq i < n$  do
3:   if  $L[i] > max$  then
4:     Update  $max$  value to  $L[i]$  // Remember that you can describe an action in words, as long as it is clear
                                   how to implement it.
5:   end if
6: end for // This is a comment in the pseudo-code. Comments should not be included in it.
7: return  $max$ 
```

---

## 2 Skip-List

### 2.1 Warm-Up and Familiarization

#### 2.1.1 Implementation of Abstract Functions

**Answer 2.1:** [Implementation in code](#)

#### 2.1.2 Analysis of the Probabilistic Process

**Answer 2.2:** The tables are:

$p = \langle \text{Insert the value of } p \text{ here} \rangle$							
$x$	$\hat{\ell}_1$	$\hat{\ell}_2$	$\hat{\ell}_3$	$\hat{\ell}_4$	$\hat{\ell}_5$	Expected Level ( $E[\ell]$ )	Average delta ( $\frac{1}{5} \cdot \sum_{i=1}^5 (\hat{\ell}_i - E[\ell])$ )
10	?	?	?	?	?	?	?
100	?	?	?	?	?	?	?
1000	?	?	?	?	?	?	?
10000	?	?	?	?	?	?	?

**Answer 2.3:**

**Answer 2.4:**

#### 2.1.3 Analysis of the operations

**Answer 2.5:** [Implementation in code](#)

**Answer 2.6:** The tables are:

$p = \langle \text{Insert the value of } p \text{ here} \rangle$			
$x$	Average Insertion	Average Search	Average Deletion
1000	?	?	?
2500	?	?	?
5000	?	?	?
10000	?	?	?
15000	?	?	?
20000	?	?	?
50000	?	?	?

**Answer 2.7:** The graph is:

**Answer 2.8:**

**Answer 2.9:**

**Answer 2.10:**

**Answer 2.11:**

## 2.2 Order Statistics

**Answer 2.12:**

## 3 Hashing

### 3.1 Introduction

#### 3.1.1 Hash Functions

Answer 3.1:

Answer 3.2:

### 3.2 Hash Implementations

Answer 3.3: [Implementation in code](#)

Answer 3.4: [Implementation in code](#)

Answer 3.5: [Implementation in code](#)

### 3.3 Hash Tables

#### 3.3.1 Introduction

Answer 3.6: [Implementation in code](#)

Answer 3.7: [Implementation in code](#)

Answer 3.8: The results are:

Linear Probing		
max $\alpha$	Average Insertion	Average Search
1/2	?	?
3/4	?	?
7/8	?	?
15/16	?	?

Answer 3.9:

Answer 3.10: The results are:

Chaining		
max $\alpha$	Average Insertion	Average Search
1/2	?	?
3/4	?	?
1	?	?
3/2	?	?
2	?	?

Answer 3.11:

**Answer 3.12:**

### **3.4 Theoretical Questions**

**Answer 3.13:**

**Answer 3.14:**

**Answer 3.15:**

**Answer 3.16:**

## 4 Designing a data structure according to given specifications

Answer 4.1:

Good Luck!