

# Data Structures- Hospital project.

Names: dina alfhaid 38490 & sara alotaibi 35491

## ❖ Project Object

In this project we want to handle the patients data (insert, delete, show in different forms) in a hospital. This data contains: the name of patient, the id of patient, admission date, whether the patient has insurance, insurance company name and department in hospital.

## ❖ Project Architecture

The data is found in a file (patients.txt) and we want to read this data from this file and represent it using three data structure Double Linked list, Array Stack and Binary search tree.

We will build a simple interactive program so the user can choose the wanted data structure from a menu and then choose the desired operation.

For each data structure there are declaration and some functions to achieve the operations we want. In the following we will discuss each data structure individually.

### 1. Doubly Linked List:

#### 1.1 Declaration:

```
struct linkNode
{
    char* name;
    char* id;
    int date;
    char* insurance;
    char* insuranceCompany;
    char* departement;
    struct linkNode *next;
    struct linkNode*prev;
};
struct linkNode *head, *tail;
char * name : A char pointer to the name's array.
```

Int id: A char pointer to the id's array.

char \* date: A char pointer to the admission's date array.

char \* insurance: A char pointer to the insurance array. Ps: true or false

char \* insuranceCompany: A char pointer to insurance's company array

char\* department: A char pointer to department's array.

struct linkNode \*previous: Previous pointer

struct linkNode \*next: Next pointer.

Then we declare the doubly linked list by defining two pointers, head and tail.

## 1.2 Functions:

```
- void CreateLinkedlist() {}
```

The function is responsible for reading the patients' data from the file and creating the linked list out of them by call the function InsertAtBeginning for each node (line here represent the node data). If the file dose'not exist it will return appropriate message.

```
- void InsertAtBeginning(char* name, int id, char* date, char* insurance, char* insuranceCompany, char* department) {}
```

This function is responsible for inserting a new patient's data to the doubly linked list. Therefore the parameters for this functions are the new patient's data.

```
- void DeleteByID(int id) {}
```

This function is responsible for deleting a patient's data from the linked list by his id. Therefor the Id is passed to the function as a parameter. The process will be to search for the node that hold this patient's id and delete the specific node.

```
- void View() {}
```

This function is responsible for showing the data of the patients in the linked list.

## 2. Array Stack

### 1.2 Declaration

```
struct stackNode
{
    char* name;
    int id;
    char* date;
    char* insurance;
    char* insuranceCompany;
    char* departement;
    struct stackNode* top;
};

struct stackNode stack;
int N; //number of elements in stack
```

We define a struct nodeStack that has the variables for the patient's data as defined in the struct node and we define a pointer to the stack and a number that holds the array stack length.

### 2.2 Functions:

- void CreateArrayStack() {}

The function read the patients' data from the file and creating the array stack out of them by calling the function push for every node in the file.

```
- void push(char* name, int id, char* date, char* insurance, char*
insuranceCompany, char* department) {}
```

This function push a new patient's data to the array stack. There for the parameters for this functions are the new patient's data.

- void pop() {}

This function the node on the top will be deleted from the array stack and returned. And the length of the stack will be decreased by one. When the stack is empty it will return appropriate message.

```
- void ViewStack() {}
```

This function show the data of the patients in the array stack. If the stack was empty it will return appropriate message.

### 3. Binary Search tree:

#### 3.1 Declaration:

```
struct treeNode {  
    char* name;  
    int id;  
    char* date;  
    char* insurance;  
    char* insuranceCompany;  
    char* departement;  
    struct treeNode* left;  
    struct treeNode* right;  
};
```

We define a struct treeNode that has the variables for the patient's data as defined in the struct treeNode and we define a pointer to the left and right of the node.

#### 2.3 Functions:

```
- void createBST() {}
```

This function create the binary search tree out of the data in the file by using insertNode function.

```
-struct treeNode* CreateNode(char* _name, int _id, char* _date, char* _insurance,  
char* _insuranceCompany, char* _department)
```

This function create node with a determine data.

```
- struct treeNode* insertNode(char* _name, int _id, char* _date, char* _insurance,  
char* _insuranceCompany, char* _department)
```

This function insert node in tree after create this node.

```
- void ViewInOrder() {}
```

This function view the data in the binary search tree performing InOrder algorithm.

```
- void ViewPreOrder() {}
```

This function view the data in the binary search tree performing PreOrder algorithm.

```
- void ViewPostOrder() {}
```

This function view the data in the binary search tree performing PostOrder algorithm.

## ❖ The content of the file:

```
Dina Alfheid,38490,1/1/2001,yes,Alhayat,surgery
Sara Alotaibi,35491,2/3/2007,false,lala,first aid
Mohammad Abdalah,203,2/3/2007,false,lala,surgery
Naya Ziad,204,2/3/2007,false,lala,first aid
Mariam Mousalli,205,4/4/2008,true,Life's Insurance,intern
Fouad Zreeq,206,5/9/2008,true,Alhayat,first aid
```

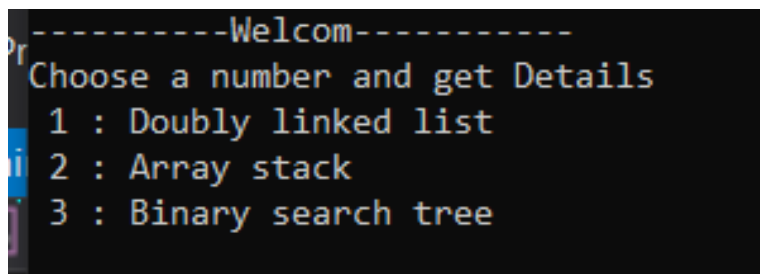
## ❖ Screen Shots & Outputs:

### 1. Main menu:

When the program starts, a menu showing a list of choices, is displayed on the console.

The choices will allow the user to chose between a number of data structures to use to achieve the required tasks.

The following screen shot shows the main menu.

A screenshot of a terminal window with a black background and white text. The text displays a welcome message and a menu of options. The first line is a separator line of dashes followed by 'Welcom' and another set of dashes. The second line prompts the user to 'Choose a number and get Details'. The following three lines list the options: '1 : Doubly linked list', '2 : Array stack', and '3 : Binary search tree'.

```
-----Welcom-----
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
```

### 2. Submenu:

After choosing a data structure a submenu will appear on the console, to allow user to perform either "add, remove, show all, back to main menu"

## 2.1 Choosing Doubly linked list: the doubly linked list will be automatically created

```
The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
```

### 2.1.1 Insert

By press 1. The program asks you to enter the data of the patient

```
-----Welcom-----
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
1

The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
1
Please enter the name of the patient
Ahmad
Please enter the id of the patient
1
Please enter the Date
12/3/2020
Does the patient have insurance?
Press 1 for yes
Press 2 for no
1
Please enter the name of the insurance company
Alhayat
Please enter the Department name
surgery
```

2.1.2 Delete: By pressing 2, you will be asked to enter the id of the patient that you want to delete

```
The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
2
Please Enter the Id of the patient you want to delete
203
Patient is deleted
*****

The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
```

If the id does not exist, it will return nothing

```
The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
2
Please Enter the Id of the patient you want to delete
3
*****

The Doubly linked list for Patients had been created
Choose a number and get the result of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
```

2.1.3 View: the list of patients will appear. Notice that the patient that we added has appeared in the list while the one we deleted did not

```
Name: Ahmad
id: 1
Date: 12/3/2020
Insurance: Yes
Insuranceid: Alhayat
Departement: surgury
.....
Name: Fouad Zreeq
id: 206
Date: 5/9/2008
Insurance: true
Insuranceid: Alhayat
Departement: first ai
.....
Name: Mariam Mousalli
id: 205
Date: 4/4/2008
Insurance: true
Insuranceid: Life's Insurance
Departement: intern
.....
Name: Naya Ziad
id: 204
Date: 2/3/2007
Insurance: false
Insuranceid: lala
Departement: first aid
.....
Name: Sara Alotaibi
id: 35491
Date: 2/3/2007
Insurance: false
Insuranceid: lala
Departement: first aid
.....
Name: Dina Alfhaid
id: 38490
Date: 1/1/2001
Insurance: yes
Insuranceid: Alhayat
Departement: surgery
.....

The Doubly linked list for Patients had been created
Choose a number and get the retult of your operation:
1 : Insert Patient
2 : Delete Patient information by id
3 : View all Patients
4 : Back to main menu
```



## 2.2 Choosing Stack Array

The stack array will be automatically created

```
The Array Stack for Patients had been created
Choose a number and get the result of your operation:
1 : Push an patient
2 : Pop the last patient
3 : View all patients
4 : Back to main menu
```

### 2.2.1 Insert:

By press 1, The programme will ask you to enter the data of the patient

```
-----Welcom-----
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
2
The Array Stack for Patients had been created
Choose a number and get the result of your operation:
1 : Push an patient
2 : Pop the last patient
3 : View all patients
4 : Back to main menu
1
Please enter the name of the patient
Dina
Please enter the id of the patient
3
Please enter the Date
2/2/2020
Does the patient has insurance?
Press 1 for yes
Press 2 for no
1
Please enter the name of the insurance company
Alhyat
Please enter the Department name
surgery
the new item was pushed successfully!
The Array Stack for Patients had been created
Choose a number and get the result of your operation:
1 : Push an patient
2 : Pop the last patient
3 : View all patients
4 : Back to main menu
```

### 2.2.2 Pop

By pressing 2, the last patient will be deleted

```
Please enter the Department name
surgery
the new item was pushed successfully!
The Array Stack for Patients had been created
Choose a number and get the result of your operation:
 1 : Push an patient
 2 : Pop the last patient
 3 : View all patients
 4 : Back to main menu
2
You choose to pop
The Array Stack for Patients had been created
Choose a number and get the result of your operation:
 1 : Push an patient
 2 : Pop the last patient
 3 : View all patients
 4 : Back to main menu
```

### 2.2.3 View

By pressing 3, we can see the menu of patients.

Notice that the patient that have deleted did appear

```
You choose to view
Name: Fouad Zreeq
  id: 206
Date: 5/9/2008
Insurance: true
Insuranceid: Alhayat
  Departement: first ai
.....
Name: Mariam Mousalli
  id: 205
Date: 4/4/2008
Insurance: true
Insuranceid: Life's Insurance
  Departement: intern
.....
Name: Naya Ziad
  id: 204
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Mohammad Abdalah
  id: 203
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: surgery
.....
Name: Sara Alotaibi
  id: 35491
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Dina Alfheid
  id: 38490
Date: 1/1/2001
Insurance: yes
Insuranceid: Alhayat
  Departement: surgery
.....
The Array Stack for Patients had been created
Choose a number and get the retult of your operation:
```

## 2.3 Choosing BST

The doubly linked list will be automatically created

### 2.3.2 View InOrder

By pressing 2 , the order of appear will be: Left, Root, Right

```
You choose to View InOrder
Name: Dina Alfheid
  id: 38490
Date: 1/1/2001
Insurance: yes
Insuranceid: Alhayat
  Departement: surgery
.....
Name: Mohammad Abdalah
  id: 203
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: surgery
.....
Name: Naya Ziad
  id: 204
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Mariam Mousalli
  id: 205
Date: 4/4/2008
Insurance: true
Insuranceid: Life's Insurance
  Departement: intern
.....
Name: Fouad Zreeq
  id: 206
Date: 5/9/2008
Insurance: true
Insuranceid: Alhayat
  Departement: first ai
.....
Name: Sara Alotaibi
  id: 35491
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
The Binary Search Tree for Patients had been created
Choose a number and get the retult of your operation:
  1 : View InOrder all patients
  2 : View PreOrder all patients
  3 : View PostOrder all patients
```

### 2.3.3 View PreOrder

By pressing 2 , the order of appear will be: Root, Left, Right

```
You choose to View PreOrder
Name: Dina Alfhaid
  id: 38490
Date: 1/1/2001
Insurance: yes
Insuranceid: Alhayat
  Departement: surgery
.....
Name: Mohammad Abdalah
  id: 203
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: surgery
.....
Name: Naya Ziad
  id: 204
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Mariam Mousalli
  id: 205
Date: 4/4/2008
Insurance: true
Insuranceid: Life's Insurance
  Departement: intern
.....
Name: Fouad Zreeq
  id: 206
Date: 5/9/2008
Insurance: true
Insuranceid: Alhayat
  Departement: first ai
.....
Name: Sara Alotaibi
  id: 35491
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
The Binary Search Tree for Patients had been created
Choose a number and get the retult of your operation:
  1 : View InOrder all patients
  2 : View PreOrder all patients
  3 : View PostOrder all patients
```

### 2.3.4 View PostOrder:

By pressing 3 , the order of appear will be: Left, Right, Root

```
You choose to View PostOrder
Name: Mohammad Abdalah
  id: 203
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: surgery
.....
Name: Naya Ziad
  id: 204
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Mariam Mousalli
  id: 205
Date: 4/4/2008
Insurance: true
Insuranceid: Life's Insurance
  Departement: intern
.....
Name: Fouad Zreeq
  id: 206
Date: 5/9/2008
Insurance: true
Insuranceid: Alhayat
  Departement: first ai
.....
Name: Sara Alotaibi
  id: 35491
Date: 2/3/2007
Insurance: false
Insuranceid: lala
  Departement: first aid
.....
Name: Dina Alfhaid
  id: 38490
Date: 1/1/2001
Insurance: yes
Insuranceid: Alhayat
  Departement: surgery
.....
The Binary Search Tree for Patients had been created
Choose a number and get the retult of your operation:
 1 : View InOrder all patients
 2 : View PreOrder all patients
 3 : View PostOrder all patients
```

if the user chose to go back to the main menu, it will appear on the console, and they "user" will be able to perform the previous steps again.

```
Choose a number and get the result of your operation:
1 : View InOrder all patients
2 : View PreOrder all patients
3 : View PostOrder all patients
4 : Back to main menu
4
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
```

If the user chose a wrong number:

```
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
7
Your input is not correct
Choose a number and get Details
1 : Doubly linked list
2 : Array stack
3 : Binary search tree
```

All changes and updates done since deliverable1:

- We implement all functions
- We add comments
- We delete the choice to create patients list, stack or BTS, because it will be created automatically when we chose one of the data structures.