

## Assignment 2

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Class Documentation</b>	<b>7</b>
4.1	AALst.AALst Class Reference . . . . .	7
4.1.1	Detailed Description . . . . .	7
4.1.2	Member Function Documentation . . . . .	7
4.1.2.1	add_stdnt() . . . . .	7
4.1.2.2	init() . . . . .	8
4.1.2.3	lst_alloc() . . . . .	8
4.1.2.4	num_alloc() . . . . .	8
4.2	DCapALst.DCapALst Class Reference . . . . .	9
4.2.1	Detailed Description . . . . .	9
4.2.2	Member Function Documentation . . . . .	9
4.2.2.1	add() . . . . .	9
4.2.2.2	capacity() . . . . .	10
4.2.2.3	elm() . . . . .	10
4.2.2.4	remove() . . . . .	10
4.3	StdntAllocTypes.DepiT Class Reference . . . . .	11

4.3.1	Detailed Description	11
4.4	StdntAllocTypes.GenT Class Reference	11
4.4.1	Detailed Description	12
4.5	SALst.SALst Class Reference	12
4.5.1	Detailed Description	12
4.5.2	Member Function Documentation	12
4.5.2.1	add()	12
4.5.2.2	allocate()	13
4.5.2.3	average()	13
4.5.2.4	elm()	14
4.5.2.5	info()	14
4.5.2.6	remove()	14
4.5.2.7	sort()	15
4.6	SeqADT.SeqADT Class Reference	15
4.6.1	Detailed Description	16
4.6.2	Constructor & Destructor Documentation	16
4.6.2.1	__init__()	16
4.6.3	Member Function Documentation	16
4.6.3.1	end()	16
4.6.3.2	next()	16
4.7	StdntAllocTypes.SInfoT Class Reference	17
4.7.1	Detailed Description	17
<b>5</b>	<b>File Documentation</b>	<b>19</b>
5.1	src/AALst.py File Reference	19
5.1.1	Detailed Description	19
5.2	src/DCapALst.py File Reference	19
5.2.1	Detailed Description	20
5.3	src/Read.py File Reference	20
5.3.1	Detailed Description	20
5.3.2	Function Documentation	20
5.3.2.1	load_dcap_data()	20
5.3.2.2	load_stdnt_data()	21
5.4	src/SALst.py File Reference	21
5.4.1	Detailed Description	21
5.5	src/SeqADT.py File Reference	21
5.5.1	Detailed Description	22
5.6	src/StdntAllocTypes.py File Reference	22
5.6.1	Detailed Description	22
<b>Index</b>		<b>23</b>

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

AALst.AALst . . . . .	7
DCapALst.DCapALst . . . . .	9
SALst.SALst . . . . .	12
SeqADT.SeqADT . . . . .	15
Enum	
StdntAllocTypes.DeptT . . . . .	11
StdntAllocTypes.GenT . . . . .	11
NamedTuple	
StdntAllocTypes.SInfoT . . . . .	17



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">AALst.AALst</a>	An abstract data type containing engineering departments and the students allocated into them	7
<a href="#">DCapALst.DCapALst</a>	An abstract data type containing the capacities of engineering departments as a list . . . . .	9
<a href="#">StdntAllocTypes.DepiT</a>	An Enumerated type of possible engineering departments . . . . .	11
<a href="#">StdntAllocTypes.GenT</a>	An Enumerated type of possible genders . . . . .	11
<a href="#">SALst.SALst</a>	An abstract data type of all first year engineering students . . . . .	12
<a href="#">SeqADT.SeqADT</a>	An abstract data type that represents a sequence of values . . . . .	15
<a href="#">StdntAllocTypes.SInfoT</a>	A NamedTuple used to represent a student . . . . .	17





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

src/ <a href="#">AALst.py</a>	
Allocation Association List Module . . . . .	19
src/ <a href="#">DCapALst.py</a>	
Department Capacity Association List . . . . .	19
src/ <a href="#">Read.py</a>	
Read . . . . .	20
src/ <a href="#">SALst.py</a>	
Student Association List . . . . .	21
src/ <a href="#">SeqADT.py</a>	
Sequence ADT . . . . .	21
src/ <a href="#">StdntAllocTypes.py</a>	
Student Allocation Types . . . . .	22



# Chapter 4

## Class Documentation

### 4.1 AALst.AALst Class Reference

An abstract data type containing engineering departments and the students allocated into them.

#### Static Public Member Functions

- def `init ()`  
*Initiazlies the [AALst](#).*
- def `add_stdnt`
- def `lst_alloc`
- def `num_alloc`

#### 4.1.1 Detailed Description

An abstract data type containing engineering departments and the students allocated into them.

#### 4.1.2 Member Function Documentation

##### 4.1.2.1 `add_stdnt()`

```
def AALst.AALst.add_stdnt (  
    dep ) [static]
```

`add_stdnt` adds a student to a specific department

#### Parameters

<i>dep</i>	A departnt of type <a href="#">StdntAllocTypes.DeptT</a>
<i>m</i>	A string representing the students macid

#### 4.1.2.2 `init()`

```
def AALst.AALst.init ( ) [static]
```

Initiazlies the [AALst](#).

The list is initialized with each department and an empty list of students for each department.

#### 4.1.2.3 `lst_alloc()`

```
def AALst.AALst.lst_alloc (
    d ) [static]
```

`lst_alloc` returns a list of students in a specific department

##### Parameters

<code>d</code>	A department of type <a href="#">StdntAllocTypes.DeptT</a>
----------------	--

##### Returns

A list of strings where each string is a macid

#### 4.1.2.4 `num_alloc()`

```
def AALst.AALst.num_alloc (
    d ) [static]
```

`num_alloc` returns the number of students in a department

##### Parameters

<code>d</code>	A department of type <a href="#">StdntAllocTypes.DeptT</a>
----------------	--

##### Returns

A integer representing the number of students in a department

The documentation for this class was generated from the following file:

- [src/AALst.py](#)

## 4.2 DCapALst.DCapALst Class Reference

An abstract data type containing the capacities of engineering departments as a list.

### Static Public Member Functions

- def [init](#) ()  
*Initializes the Department Capacity List to be empty.*
- def [add](#)  
*Adds a department and its capacity to the list.*
- def [remove](#)  
*Removes a department and its capacity from the list*
- def [elm](#)  
*elm checks if a department has been added*
- def [capacity](#)  
*capacity returns the capacity of a department*

### 4.2.1 Detailed Description

An abstract data type containing the capacities of engineering departments as a list.

### 4.2.2 Member Function Documentation

#### 4.2.2.1 add()

```
def DCapALst.DCapALst.add (
    d ) [static]
```

Adds a department and its capacity to the list.

#### Exceptions

<i>throws</i>	KeyError if the given department has been added before
---------------	--

#### Parameters

<i>d</i>	A department of type <a href="#">StdntAllocTypes.DeptT</a>
<i>n</i>	An integer representing the capacity of the department (d parameter)

#### 4.2.2.2 capacity()

```
def DCapALst.DCapALst.capacity (
    d ) [static]
```

capacity returns the capacity of a department

##### Exceptions

<i>throws</i>	KeyError if the department given is not in <a href="#">DCapALst</a>
---------------	---

##### Parameters

<i>d</i>	A department of type <a href="#">StdntAllocTypes.DeptT</a>
----------	--

##### Returns

An integer representing the capacity of the department given as a parameter.

#### 4.2.2.3 elm()

```
def DCapALst.DCapALst.elm (
    d ) [static]
```

elm checks if a department has been added

##### Parameters

<i>d</i>	A department of type <a href="#">StdntAllocTypes.DeptT</a>
----------	--

##### Returns

True if the department has been added, otherwise False

#### 4.2.2.4 remove()

```
def DCapALst.DCapALst.remove (
    d ) [static]
```

Removes a department and its capacity from the list

##### Exceptions

<i>throws</i>	KeyError if the given department is not in <a href="#">DCapALst</a>
---------------	---

## Parameters

<i>d</i>	A department of type <a href="#">StdntAllocTypes.DeptT</a> to be removed
----------	--

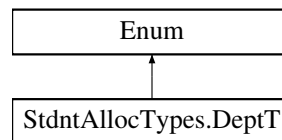
The documentation for this class was generated from the following file:

- [src/DCapALst.py](#)

## 4.3 StdntAllocTypes.DeptT Class Reference

An Enumerated type of possible engineering departments.

Inheritance diagram for StdntAllocTypes.DeptT:



### Static Public Attributes

- string **civil** = "civil"
- string **chemical** = "chemical"
- string **electrical** = "electrical"
- string **mechanical** = "mechanical"
- string **software** = "software"
- string **materials** = "materials"
- string **engphys** = "engphys"

### 4.3.1 Detailed Description

An Enumerated type of possible engineering departments.

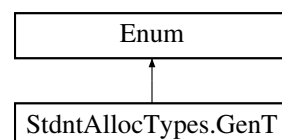
The documentation for this class was generated from the following file:

- [src/StdntAllocTypes.py](#)

## 4.4 StdntAllocTypes.GenT Class Reference

An Enumerated type of possible genders

Inheritance diagram for StdntAllocTypes.GenT:



## Static Public Attributes

- string **male** = "male"
- string **female** = "female"

### 4.4.1 Detailed Description

An Enumerated type of possible genders

The documentation for this class was generated from the following file:

- [src/StdntAllocTypes.py](#)

## 4.5 SALst.SALst Class Reference

An abstract data type of all first year engineering students.

### Static Public Member Functions

- def [init](#) ()  
*init initializes the list of students to be empty*
- def [add](#)  
*Adds a student into the [SALst](#).*
- def [remove](#)  
*Removes a student from the [SALst](#).*
- def [elm](#)  
*elm checks if a student is already in the [SALst](#)*
- def [info](#)  
*returns the information assoociated with a student*
- def [sort](#)  
*Sorts a subset of students based on GPA.*
- def [average](#)  
*Computes the average of a particular subset of students.*
- def [allocate](#) ()  
*Allocates students in [SALst](#) into their program*

### 4.5.1 Detailed Description

An abstract data type of all first year engineering students.

### 4.5.2 Member Function Documentation

#### 4.5.2.1 [add\(\)](#)

```
def SALst.SALst.add (
    m ) [static]
```

Adds a student into the [SALst](#).



## Exceptions

<i>throws</i>	KeyError if the student given has been added before
---------------	---

## Parameters

<i>m</i>	A string of a student's macid
<i>i</i>	Information of a student given with the data type <a href="#">StdntAllocTypes.SInfoT</a>

## 4.5.2.2 allocate()

```
def SALst.SALst.allocate ( ) [static]
```

Allocates students in [SALst](#) into their program

Students are allocated into a department in AALst. Students with free choice are allocated first. The remaining students are allocated in a order based on their GPA, a student is allocated into their highest preferred choice that is not full in capacity.

## Exceptions

<i>throws</i>	RuntimeError if all of a student's choices are full.
---------------	--

## 4.5.2.3 average()

```
def SALst.SALst.average (
    f ) [static]
```

Computes the average of a particular subset of students.

The method is given a function that is able to filter a student. The function takes in a student(SInfoT) and returns True if they pass the filter. The method will then compute the average GPA amongst students who passed the filter.

## Exceptions

<i>throws</i>	ValueError if there are no students that pass the filter function.
---------------	--

## Parameters

<i>f</i>	A filtering function that returns a boolean
----------	---

**Returns**

A float representing the average GPA amongst a subset of students

**4.5.2.4 elm()**

```
def SALst.SALst.elm (
    m ) [static]
```

elm checks if a student is already in the [SALst](#)

**Parameters**

<i>m</i>	A string of a student's macid
----------	-------------------------------

**Returns**

True if a student is in [SALst](#), otherwise False

**4.5.2.5 info()**

```
def SALst.SALst.info (
    m ) [static]
```

returns the information assoociated with a student

**Exceptions**

<i>throws</i>	KeyError if the student is not found
---------------	--------------------------------------

**Parameters**

<i>m</i>	A string of a student's macid
----------	-------------------------------

**Returns**

A students information with the type [StdntAllocTypes.SInfoT](#)

**4.5.2.6 remove()**

```
def SALst.SALst.remove (
    m ) [static]
```

Removes a student from the [SALst](#).

## Exceptions

<i>throws</i>	KeyError if a student to be removed is not found
---------------	--

## Parameters

<i>m</i>	A string of a student's macid
----------	-------------------------------

## 4.5.2.7 sort()

```
def SALst.SALst.sort (
    f ) [static]
```

Sorts a subset of students based on GPA.

The method is given a function that is able to filter a student. The filter function takes in a student (SInfoT) and returns True if they pass the filter. The method will return a list of macids that passed the filter, sorted by their GPA in descending order.

## Parameters

<i>f</i>	A filtering function that returns a boolean
----------	---

## Returns

A list of strings (each string is a macid) sorted by their GPA in descending order

The documentation for this class was generated from the following file:

- [src/SALst.py](#)

## 4.6 SeqADT.SeqADT Class Reference

An abstract data type that represents a sequence of values.

## Public Member Functions

- `def __init__`  
*SeqADT constructor.*
- `def start (self)`  
*start will reset the index state variable to 0*
- `def next (self)`  
*next will return the next value in the sequence*
- `def end (self)`  
*end will check if there are more items in the sequence*

### 4.6.1 Detailed Description

An abstract data type that represents a sequence of values.

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 `__init__()`

```
def SeqADT.SeqADT.__init__ (
    self,
    x )
```

[SeqADT](#) constructor.

Initializes the state variables of [SeqADT](#). The state variables are a list that is given as a parameter and a variable used to index the list (initialized to 0).

#### Parameters

<i>x</i>	A list of values
----------	------------------

### 4.6.3 Member Function Documentation

#### 4.6.3.1 `end()`

```
def SeqADT.SeqADT.end (
    self,
    bool )
```

`end` will check if there are more items in the sequence

#### Returns

True if there are no more items in the sequence, otherwise False

#### 4.6.3.2 `next()`

```
def SeqADT.SeqADT.next (
    self )
```

`next` will return the next value in the sequence

### Exceptions

<i>throws</i>	StopIteration if there is no more items in the sequence
---------------	---

### Returns

value of next item in the sequence

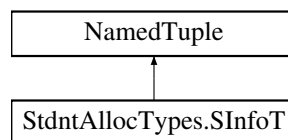
The documentation for this class was generated from the following file:

- [src/SeqADT.py](#)

## 4.7 StdntAllocTypes.SInfoT Class Reference

A NamedTuple used to represent a student.

Inheritance diagram for StdntAllocTypes.SInfoT:



### 4.7.1 Detailed Description

A NamedTuple used to represent a student.

A student has a: first name, last name, gender (given as a [GenT](#) type), gpa, sequence of departments(given as a SeqADT of [DeptT](#)'s), and a boolean to represent if they have free choice.

The documentation for this class was generated from the following file:

- [src/StdntAllocTypes.py](#)



## Chapter 5

# File Documentation

### 5.1 src/AALst.py File Reference

Allocation Association List Module

#### Classes

- class [AALst.AALst](#)

*An abstract data type containing engineering departments and the students allocated into them.*

#### 5.1.1 Detailed Description

Allocation Association List Module

#### Author

Dominik Buszowiecki

#### Date

February 9, 2019

### 5.2 src/DCapALst.py File Reference

Department Capacity Association List

#### Classes

- class [DCapALst.DCapALst](#)

*An abstract data type containing the capacities of engineering departments as a list.*

### 5.2.1 Detailed Description

Department Capacity Association List

Author

Dominik Buszowiecki

Date

February 9, 2019

## 5.3 src/Read.py File Reference

Read

### Functions

- def [Read.load\\_stdnt\\_data](#)  
*Loads students from a file into the SALst.*
- def [Read.load\\_dcap\\_data](#)  
*Loads department capacities from a file into the DCapALst.*

### 5.3.1 Detailed Description

Read

Author

Dominik Buszowiecki

Date

February 9, 2019

### 5.3.2 Function Documentation

#### 5.3.2.1 load\_dcap\_data()

```
def Read.load_dcap_data (
    s )
```

Loads department capacities from a file into the DCapALst.

Each line in the file represents a department. The format of each line should be:  
department\_name, capacity  
where capacity is an integer.



## Parameters

s	A string representing the name of the file
---	--

## 5.3.2.2 load\_stdnt\_data()

```
def Read.load_stdnt_data (
    s )
```

Loads students from a file into the SALst.

Each line in the file represents a student. The format of each line should be: macid, firstname, lastname, gender, gpa, [choice1, choice2, ...], freechoice where gpa is a real number, gender is either male or female and freechoice is either True or False.

## Parameters

s	A string representing the name of the file
---	--

## 5.4 src/SALst.py File Reference

Student Association List

## Classes

- class [SALst.SALst](#)  
*An abstract data type of all first year engineerng students.*

## 5.4.1 Detailed Description

Student Association List

## Author

Dominik Buszowiecki

## Date

February 9, 2019

## 5.5 src/SeqADT.py File Reference

Sequence ADT

## Classes

- class [SeqADT.SeqADT](#)  
*An abstract data type that represents a sequence of values.*

### 5.5.1 Detailed Description

Sequence ADT

Author

Dominik Buszowiecki

Date

February 9, 2019

## 5.6 src/StdntAllocTypes.py File Reference

Student Allocation Types

## Classes

- class [StdntAllocTypes.GenT](#)  
*An Enumerated type of possible genders*
- class [StdntAllocTypes.DeptT](#)  
*An Enumerated type of possible engineering departments.*
- class [StdntAllocTypes.SInfoT](#)  
*A NamedTuple used to represent a student.*

### 5.6.1 Detailed Description

Student Allocation Types

Author

Dominik Buszowiecki

Date

February 9, 2019

# Index

- `__init__`
    - `SeqADT::SeqADT`, [16](#)
- `AALst.AALst`, [7](#)
- `AALst::AALst`
  - `add_stdnt`, [7](#)
  - `init`, [8](#)
  - `lst_alloc`, [8](#)
  - `num_alloc`, [8](#)
- `add`
  - `DCapALst::DCapALst`, [9](#)
  - `SALst::SALst`, [12](#)
- `add_stdnt`
  - `AALst::AALst`, [7](#)
- `allocate`
  - `SALst::SALst`, [13](#)
- `average`
  - `SALst::SALst`, [13](#)
- `capacity`
  - `DCapALst::DCapALst`, [9](#)
- `DCapALst.DCapALst`, [9](#)
- `DCapALst::DCapALst`
  - `add`, [9](#)
  - `capacity`, [9](#)
  - `elm`, [10](#)
  - `remove`, [10](#)
- `elm`
  - `DCapALst::DCapALst`, [10](#)
  - `SALst::SALst`, [14](#)
- `end`
  - `SeqADT::SeqADT`, [16](#)
- `info`
  - `SALst::SALst`, [14](#)
- `init`
  - `AALst::AALst`, [8](#)
- `load_dcap_data`
  - `Read.py`, [20](#)
- `load_stdnt_data`
  - `Read.py`, [21](#)
- `lst_alloc`
  - `AALst::AALst`, [8](#)
- `next`
  - `SeqADT::SeqADT`, [16](#)
- `num_alloc`
  - `AALst::AALst`, [8](#)
- `Read.py`
  - `load_dcap_data`, [20](#)
  - `load_stdnt_data`, [21](#)
- `remove`
  - `DCapALst::DCapALst`, [10](#)
  - `SALst::SALst`, [14](#)
- `SALst.SALst`, [12](#)
- `SALst::SALst`
  - `add`, [12](#)
  - `allocate`, [13](#)
  - `average`, [13](#)
  - `elm`, [14](#)
  - `info`, [14](#)
  - `remove`, [14](#)
  - `sort`, [15](#)
- `SeqADT.SeqADT`, [15](#)
- `SeqADT::SeqADT`
  - `__init__`, [16](#)
  - `end`, [16](#)
  - `next`, [16](#)
- `sort`
  - `SALst::SALst`, [15](#)
- `src/AALst.py`, [19](#)
- `src/DCapALst.py`, [19](#)
- `src/Read.py`, [20](#)
- `src/SALst.py`, [21](#)
- `src/SeqADT.py`, [21](#)
- `src/StdntAllocTypes.py`, [22](#)
- `StdntAllocTypes.DeptT`, [11](#)
- `StdntAllocTypes.GenT`, [11](#)
- `StdntAllocTypes.SInfoT`, [17](#)