

## SC1003 Assignment

The NTU Name Card Holder has a capacity of MAX (e.g., 5 for this program) name cards. You are required to write a NTU Name Card Holder Management Program. The program uses an array of MAX structures for the name card holder.

Each structure should hold:

- nameCardID – It stores the name card identification number.
- personName – It stores the name of the person in the name card.
- companyName – It stores the name of the person's company in the name card.

A structure is defined to represent a name card record as follows:

```
typedef struct
{
    int nameCardID;
    char personName[20];
    char companyName[20];
} NameCard;
```

The **nameCardID** must be **unique**. Therefore, each NameCard structure will have a different number of nameCardID when it is created. You may assume that the name of a person or company is not more than 20 characters long. In the program, it should display a menu to support the following functions:

NTU NAME CARD HOLDER MANAGEMENT PROGRAM:

- 1: listNameCards()
- 2: addNameCard()
- 3: removeNameCard()
- 4: getNameCard()
- 5: quit

The program should execute the functions (or options) of its menu. It will continue execution until the user selects to quit from the program.

The functions are described as follows:

(1) listNameCards() - The function requirements are given below:

- The function prints the following message when it is executed:  
**"listNameCards():"**
- It prints the content of each name card in the name card holder.
- If the name card holder is empty, the function should display the message:  
**"The name card holder is empty"**

(2) addNameCard() - The function requirements are given below:

- The function prints the following message when it is executed:  
**"addNameCard():"**

- It adds a new name card into the name card holder. Before and after this function call, the name card holder is an array of NameCard structures stored in **ascending order** according to the integer value of nameCardID.
- After adding the new name card, the function should display the message:  
**"The name card has been added successfully"**
- If the name card's nameCardID has existed in the name card holder, the function should display the message:  
**"The nameCardID has already existed"**  
The program will then ask the user select a new option for execution.
- If name card holder is full before insertion, i.e., it is equal to the maximum capacity of MAX (e.g., 5), the function should display the message:  
**"The name card holder is full"**  
and no addition should be done.

(3) removeNameCard() - The function requirements are given below:

- The function prints the following message when it is executed:  
**"removeNameCard():"**
- It removes the **first appearance** of the target person name of the name card from the name card holder.
- The string matching between the target and person name of the name card stored in the name card holder should be done irregardless of the letter cases (upper case letter and lower case letter will be treated to be the same). As such, when doing target string matching, all characters in the string should be converted into lower case or upper case before matching.
- Before and after this function call, the name card holder is an array of structures stored in **ascending order** according to the integer value of nameCardID.
- After the name card is removed, the function should display the message:  
**"The name card is removed"**  
and the contents of the corresponding name card should be displayed.
- If the name card holder is empty before removal, the function should display the message:  
**"The name card holder is empty"**  
and no removal should be done.
- If target person name does not exist in the name card holder, the function should display the error message:  
**"The target person name is not in the name card holder"**

(4) getNameCard() - The function requirements are given below:

- The function prints the following message when it is executed:  
**"getNameCard():"**
- It finds the **first appearance** of the target person name of the name card stored in the name card holder, and prints the target name card information to the screen.
- The target string matching should be done irregardless of the letter cases (upper case letter and lower case letter will be treated to be the same).
- If the target person name is found in the name card holder, the function should display the message:  
**"The target person name is found"**  
and display the contents of the corresponding name card.

- If the target person name does not exist in the name card holder, the function should issue the message:  
**"The target person name is not found"**

You are required to write the program and the functions according to the program requirements. Note that:

- You should design the program and define the functions according to the requirements.
- You only need to consider the requirements stated in the program specification, and you do not need to implement any user input checking which is not stated in the specification.
- You may add any other supporting functions in the program if needed.
- You may include any C library functions in your program if needed.
- Sample test case are given below.

A sample program running session is given below (please note that the input data are shown in orange color):

```
NTU NAME CARD HOLDER MANAGEMENT PROGRAM:
1: listNameCards()
2: addNameCard()
3: removeNameCard()
4: getNameCard()
5: quit
Enter your choice:
1
listNameCards():
The ncHolder is empty
Enter your choice:
2
addNameCard():
Enter nameCardID:
1
Enter personName:
SC Hui
Enter companyName:
NTU
The name card has been added successfully
Enter your choice:
2
addNameCard():
Enter nameCardID:
9
Enter personName:
PC Leong
Enter companyName:
NTU
The name card has been added successfully
Enter your choice:
1
listNameCards():
nameCardID: 1
personName: SC Hui
```

```

companyName: NTU
nameCardID: 9
personName: PC Leong
companyName: NTU
Enter your choice:
5

```

The test cases for the program are given below. The test cases are organized in the form of input and output data for the convenience of your testing in APAS.

### Sample Test Cases (Pretest)

Case 1 - listNameCard – empty

Input	Output
1 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: <b>listNameCards() :</b> <b>The name card holder is empty</b> Enter your choice:

Case 2 - addNameCard (3 cards) + listNameCard

Input	Output
2 9 PC Leong NTU 2 5 WC Tan NTU 2 7 Philip Fu NTU 1 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard() : Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: <b>addNameCard() :</b> Enter nameCardID: Enter personName: Enter companyName: <b>The name card has been added successfully</b> Enter your choice: addNameCard() : Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: listNameCards() : nameCardID: 5 personName: WC Tan companyName: NTU

	nameCardID: 7 personName: Philip Fu companyName: NTU nameCardID: 9 personName: PC Leong companyName: NTU Enter your choice:
--	---

**Case 3 - addNameCard (3 cards and existed) + listNameCard**

Input	Output
2 9 PC Leong NTU 2 5 WC Tan NTU 2 7 Philip Fu NTU 1 2 5 Peter Chan SMU 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: listNameCards(): nameCardID: 5 personName: WC Tan companyName: NTU nameCardID: 7 personName: Philip Fu companyName: NTU nameCardID: 9 personName: PC Leong companyName: NTU Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: <b>The nameCardID has already existed</b> Enter your choice:

**Case 4 - addNameCard (5 cards) + listNameCard**

Input	Output
2 1	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards()

SC Hui	2: addNameCard()
NTU	3: removeNameCard()
2	4: getNameCard()
9	5: quit
PC Leong	Enter your choice:
NTU	addNameCard():
2	Enter nameCardID:
5	Enter personName:
WC Tan	Enter companyName:
NTU	The name card has been added successfully
2	Enter your choice:
7	addNameCard():
Philip Fu	Enter nameCardID:
NTU	Enter personName:
2	Enter companyName:
3	The name card has been added successfully
Daniel Foo	Enter your choice:
SMU	addNameCard():
1	Enter nameCardID:
5	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	listNameCards():
	nameCardID: 1
	personName: SC Hui
	companyName: NTU
	nameCardID: 3
	personName: Daniel Foo
	companyName: SMU
	nameCardID: 5
	personName: WC Tan
	companyName: NTU
	nameCardID: 7
	personName: Philip Fu
	companyName: NTU
	nameCardID: 9
	personName: PC Leong
	companyName: NTU
	Enter your choice:

**Case 5 - addNameCard (full condition) + listNameCard**

Input	Output
2	NTU NAME CARD HOLDER MANAGEMENT PROGRAM:
1	1: listNameCards()
SC Hui	2: addNameCard()
NTU	3: removeNameCard()

2	4: getNameCard()
9	5: quit
PC Leong	Enter your choice:
NTU	addNameCard():
2	Enter nameCardID:
5	Enter personName:
WC Tan	Enter companyName:
NTU	The name card has been added successfully
2	Enter your choice:
7	addNameCard():
Philip Fu	Enter nameCardID:
NTU	Enter personName:
2	Enter companyName:
3	The name card has been added successfully
Daniel Foo	Enter your choice:
SMU	addNameCard():
1	Enter nameCardID:
2	Enter personName:
11	Enter companyName:
Kenny Tay	The name card has been added successfully
SMU	Enter your choice:
5	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	listNameCards():
	nameCardID: 1
	personName: SC Hui
	companyName: NTU
	nameCardID: 3
	personName: Daniel Foo
	companyName: SMU
	nameCardID: 5
	personName: WC Tan
	companyName: NTU
	nameCardID: 7
	personName: Philip Fu
	companyName: NTU
	nameCardID: 9
	personName: PC Leong
	companyName: NTU
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	<b>The name card holder is full</b>
	Enter your choice:

#### Case 6 - removeNameCard (empty)

Input	Output
1 3 SC Hui 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: listNameCards(): <b>The name card holder is empty</b> Enter your choice: removeNameCard(): Enter personName: <b>The name card holder is empty</b> Enter your choice:

#### Case 7 - removeNameCard (exact string matching & found removed)

Input	Output
2 9 PC Leong NTU 2 5 WC Tan NTU 2 3 Daniel Foo SMU 3 WC Tan 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: removeNameCard(): Enter personName: <b>The name card is removed</b> nameCardID: 5 personName: WC Tan companyName: NTU Enter your choice:

#### Case 8 - removeNameCard (mixed case string matching & found removed)

Input	Output
2 9	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards()



PC Leong NTU 2 5 WC Tan NTU 2 3 Daniel Foo SMU 3 Wc tan 5	2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: removeNameCard(): Enter personName: <b>The name card is removed</b> nameCardID: 5 personName: WC Tan companyName: NTU Enter your choice:
---	---

#### Case 9 - removeNameCard (not found)

Input	Output
2 9 PC Leong NTU 2 5 WC Tan NTU 2 3 Daniel Foo SMU 3 Pcc Leong 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: removeNameCard():

	Enter personName: <b>The target person name is not in the name card holder</b> Enter your choice:
--	---

#### Case 10 - removeNameCard (duplicate person name)

Input	Output
2	NTU NAME CARD HOLDER MANAGEMENT PROGRAM:
9	1: listNameCards()
PC Leong	2: addNameCard()
NTU	3: removeNameCard()
2	4: getNameCard()
5	5: quit
WC Tan	Enter your choice:
NTU	addNameCard():
2	Enter nameCardID:
3	Enter personName:
PC Leong	Enter companyName:
SMU	The name card has been added successfully
1	Enter your choice:
3	addNameCard():
PC Leong	Enter nameCardID:
<del>SMU</del>	Enter personName:
5	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	listNameCards():
	nameCardID: 3
	personName: PC Leong
	companyName: SMU
	nameCardID: 5
	personName: WC Tan
	companyName: NTU
	nameCardID: 9
	personName: PC Leong
	companyName: NTU
	Enter your choice:
	removeNameCard():
	Enter personName:
	<b>The name card is removed</b>
	nameCardID: 3
	personName: PC Leong
	companyName: SMU
	Enter your choice:

#### Case 11 - getNameCard (found)

Input	Output
2	NTU NAME CARD HOLDER MANAGEMENT PROGRAM:
9	1: listNameCards()
PC Leong	2: addNameCard()
NTU	3: removeNameCard()
2	4: getNameCard()

5 WC Tan NTU 2 3 PC Tan SMU 4 WC tan 5	5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: getNameCard(): Enter personName: <b>The target person name is found</b> nameCardID: 5 personName: WC Tan companyName: NTU Enter your choice:
---	--

#### Case 12 - getNameCard (not found)

Input	Output
2 9 PC Leong NTU 2 5 WC Tan NTU 2 3 PC Tan SMU 4 pcc Leong 5	NTU NAME CARD HOLDER MANAGEMENT PROGRAM: 1: listNameCards() 2: addNameCard() 3: removeNameCard() 4: getNameCard() 5: quit Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: addNameCard(): Enter nameCardID: Enter personName: Enter companyName: The name card has been added successfully Enter your choice: getNameCard(): Enter personName: <b>The target person name is not found</b> Enter your choice:

--	--

**Case 13 - getNameCard (unique id with duplicated name)**

Input	Output
2	NTU NAME CARD HOLDER MANAGEMENT PROGRAM:
9	1: listNameCards()
PC Leong	2: addNameCard()
NTU	3: removeNameCard()
2	4: getNameCard()
5	5: quit
WC Tan	Enter your choice:
NTU	addNameCard():
2	Enter nameCardID:
3	Enter personName:
PC Leong	Enter companyName:
SMU	The name card has been added successfully
4	Enter your choice:
PC Leong	addNameCard():
5	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	addNameCard():
	Enter nameCardID:
	Enter personName:
	Enter companyName:
	The name card has been added successfully
	Enter your choice:
	getNameCard():
	Enter personName:
	<b>The target person name is found</b>
	nameCardID: 3
	personName: PC Leong
	companyName: SMU
	Enter your choice: