

# Phone Number Storage

## Task 1

Define a C++11 enumeration for the type of a phone number with five possible values: at home, at work, fax, mobile and other.

## Task 2

Define a C/C++ structure for a contact with two C++ strings for phone number and name as components as well as the type of the contact defined in enumeration in task 1 (not of type `char` or `int` or `string` or...).

Define a global integer constant for the maximum number of contacts (length of array) initialised with a value  $> 1$ , a global array of contacts with above constant fix length of above defined C++ structure as well as a global integer variable with the actual number of stored contacts initialised with 0.

## Task 3

Define a function without parameter and Boolean return value reading the data for one contact (the name shall consist of several words like last name and first name, i.e. read a whole input line; see example below) from standard character input stream and insert it sorted by name into the array (to keep it simple several same name contact entries are possible). Return `true`, if this succeeds, or `false`, if storing fails e.g. due to a completely full array.

## Task 4

Define a function without parameter and a Boolean return value reading a name from standard character input stream, search for a contact with this name in the array and delete it (any occurrence of the same exact name). Return `true`, if the deletion succeeds, or `false`, if e.g. the name is not found in the array.

## Task 5

Define a function without parameter and no return value writing all contacts sorted alphabetically by name formatted as a table with three columns onto standard character output stream (use C++ output manipulators, not function `printf`; see example below).

## Task 6

Define a `main` function with a small menu, that you can choose to input a new contact, to delete a contact, to print or to delete all contacts as well as for the end of program. Call your functions above, do not input a name, phone number or type of the number directly in `main` itself. In case of the called function returns value `true` give a positive feedback and `false` a respective error message (see example below).

### Important to Regard

Only use C++ input and output, **no calls** of `scanf` or `printf` function, your source code file has to have name `h1_<your matriculation number>.cpp`, the ending `.cpp` is essential for the plagiarism checker as well as for our check programs, **no text file** with ending `.txt`, **no project file** `.cbp` or similar and **no** `.rar`, `.zip`, ... **file** (the plagiarism checker does not understand all these formats, therefore you would get 0 points for it), also your program is **only allowed containing ASCII characters**, **only includes of standard C++ libraries** (i.e. no libraries with ending `.h` like `conio.h`, `stdio.h`, `windows.h`, ... or `.hpp`) and shall follow standard **C++11** (or newer).

### Example Program Run:

```
*** 0 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
1
please enter phone number: 001 555 555123
please enter name: Adams, Scott
type of number (f for fax, h for home, m for mobile, w for work, o for other): w
contact stored
*** 1 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
1
please enter phone number: +49 203 379 5555
please enter name: Dilbert in Duisburg
type of number (f for fax, h for home, m for mobile, w for work, o for other): o
contact stored
*** 2 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
1
please enter phone number: 089 123 456
please enter name: Musterfrau, Erika
type of number (f for fax, h for home, m for mobile, w for work, o for other): h
contact stored
*** 3 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
3
Adams, Scott          001 555 555123          (work)
Dilbert in Duisburg  +49 203 379 5555        (other)
Musterfrau, Erika    089 123 456            (home)
```

```

*** 3 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
1
please enter phone number: 987654
please enter name: A1 Service
type of number (f for fax, h for home, m for mobile, w for work, o for other): f
contact stored
*** 4 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
3
A1 Service                987654                (fax)
Adams, Scott              001 555 555123        (work)
Dilbert in Duisburg      +49 203 379 5555      (other)
Musterfrau, Erika        089 123 456           (home)
*** 4 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
2
please enter name to delete: Dilbert
contact not found
*** 4 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
3
A1 Service                987654                (fax)
Adams, Scott              001 555 555123        (work)
Dilbert in Duisburg      +49 203 379 5555      (other)
Musterfrau, Erika        089 123 456           (home)
*** 4 contact entries ***

```

```

0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
2
please enter name to delete: Dilbert in Duisburg
contact deleted
*** 3 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
3
A1 Service          987654          (fax)
Adams, Scott        001 555 555123      (work)
Musterfrau, Erika   089 123 456         (home)
*** 3 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
2
please enter name to delete: A1 Service
contact deleted
*** 2 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
3
Adams, Scott        001 555 555123      (work)
Musterfrau, Erika   089 123 456         (home)
*** 2 contact entries ***
0 end
1 new contact
2 delete contact
3 show all contacts
9 empty contact memory
9
all contacts deleted

```

\*\*\* 0 contact entries \*\*\*

0 end

1 new contact

2 delete contact

3 show all contacts

9 empty contact memory

3

\*\*\* 0 contact entries \*\*\*

0 end

1 new contact

2 delete contact

3 show all contacts

9 empty contact memory

0

Process returned 0 (0x0) execution time : 4.328 s

Press any key to continue.