

1. Complete the header file named **main.h**: define the following structs. The first one is done for you. You should look at the function definitions in **main.cpp** to see what *data members names* you must use.

Name	Contains
name_t	first and last names example: Philo McGiffin
date_t	month, day, year (int) example: 7 4 1776
tele_t	phone # in 3 parts: area code, exchange, number (int) example: 215 323 1367
midn_t	name, alpha code, company, major, date of birth, phone # (alpha code and company are int, major is string)

2. Complete **main.cpp** by writing the body of these functions:

```
// returns true if a < b, otherwise returns false
bool before( int a, int b );

// returns true if a comes before b in ASCII order, otherwise false
bool before( string a, string b );

// returns true if a occurred before b, otherwise returns false
bool before( date_t a, date_t b );
```

3. Complete the body of this function in **main.cpp** by replacing what is in red with a proper call to function **before()**:

```
int findMin( midn_t* mids, int n, int from, int field )
{
    int index_of_min = from;
    midn_t smallest = mids[index_of_min];

    for( int i = from+1; i < n; i++ )
    {
        bool isbefore;
        switch( field )
        {
            case 1: isbefore = false; break; // sort on last name
            case 2: isbefore = false; break; // sort on alpha code
            case 3: isbefore = false; break; // sort on company
            case 4: isbefore = false; break; // sort on major
            case 5: isbefore = false; break; // sort on date of birth
        }
        if( isbefore )
        {
            smallest = mids[i];
            index_of_min = i;
        }
    }
    return index_of_min;
}
```

4. Turn in: (1) This sheet with your name filled in, to which is stapled ...  
 (2) A consistently formatted hardcopy printout of **main.h** and **main.cpp**  
 (3) Screen snapshots showing file **midn\_data.txt** sorted on each field (0 -> 4).

(NOTE: an example solution executable is provided)