

Exercise 4:

Submit your solutions (source code) to the questions below by email to your instructor and TA(s) by Monday, October 29th (16:30).

Question 1: Const-correctness (25 points).

Download the files [Date.h](#), [Date.cpp](#) and [test_date.cpp](#). Modify Date.h such that the class is const-correct, i.e. arguments that are not modified by a method should be marked as const, and methods that do not modify the object should be marked as const. Complete Date.cpp such that all methods in Date.h are implemented. A brief description of the methods is given below. You can use the file test_date.cpp, to test your modified class.

List and short description of the methods of Date:

- set() takes three arguments corresponding to the month, day and year and use these values to set its internal data (month, day and year)
- print() prints out the date stored in the object
- get_month(), get_day(), get_year() return respectively the month, the day and the year stored in a Date object

Question 2: Inline functions (25 points).

Create two files: "utils.h" and "test_utils.cpp". In "utils.h" define the following functions as inline:

- `cube()`: takes as argument an int, and returns an int equal to the cube of its argument
- `dist()`: takes as arguments two arrays of int (of same length) and their length and returns the Euclidean distance between these arrays (i.e. the square root of the sum of the squared difference between each term)

In "test_utils.cpp", type (and complete) the following code:

```
// test_utils.cpp

#include <cassert>
#include <cmath>
// Include all other necessary headers here

int main(void) {
    int num = 2;
    int num3 = cube(num);
    assert(num3==8);

    num = 5;
    num3 = cube(num);
    assert(num3==125);
}
```

```
int v1[2] = {0, 0};
int v2[2] = {1, 1};
double d = dist(v1, v2, 2);
assert(d==sqrt(2.0));

int v3[5] = {1, 1, 2, 1, 3};
int v4[5] = {5, 2, 1, 4, 1};
d = dist(v3, v4, 5);
assert(fabs(d-5.56776)<=1e-5);

return 0;
}
```

Question 3: Inline methods (25 points).

Modify the files from the question 1 (Date.h and/or Date.cpp) such that the methods: set(), print(), get_month(), get_day() and get_year() are indicated as inline. There are two ways to do it: you are free to use either of these methods.

Question 4: Is an object const? (25 points)

Define a class Constness with a method is_const() that returns true if the object receiver is const and false otherwise. In order to do that consider overloading is_const(). You can test

your solution with the following program:

```
#include <cassert>
#include "Constness.h"

void pass_const(const Constness& o) {
    assert(o.is_const() == true);
}

void pass_ref(Constness& o) {
    assert(o.is_const() == false);
}

void pass_normal(Constness o) {
    assert(o.is_const() == false);
}

int main(void) {
    Constness o;
    pass_const(o);
    pass_ref(o);
    pass_normal(o);

    const Constness const_obj = o;
    assert(const_obj.is_const() == true);

    return 0;
}
```

About overloading: we will study overloading in details in [week 5](#) (and also in [week 9](#) for operator overloading). For this exercise, all you need to know about overloading is that in C++ you can define several functions or methods with the same name as long as their signature (or prototype) is different.