Lab 1 – C++ Primer

- C++ review
- Complexity
- Recursion

Problem 1

Write a function to capitalize a string, using the following prototype:

```
char* upper(char *arr, int n)
```

For example, doing this:

```
char arr[6] = {'a', 'B', 'c', 'd', '3', '\0'};
cout << arr << endl;
upper(arr, 5);
cout << arr;
```

Should print:

```
aBcd3
ABCD3
```

Problem 2

Write a function that multiply two **double** matrices of size M x N and N x P. The multiplication result is then returned at the end of the function through a pointer.

Notes:

- M, N and P are arbitrary positive integers.
- The multiplication result between two matrices of size M x N and N x P is another matrix of size M x P.

Problem 3

Fibonacci words are defined as follow:

$$S_0 = 0$$

$$S_1 = 01$$

$$S_n = [S_{n-1}, S_{n-2}]$$

For example, $S_2 = [S_1, S_0] = [01,0] = 010$, $S_3 = [S_2, S_1] = 01001$.

Write a function that takes a non-negative integer n as input and prints the n-th Fibonacci word (n starts at 0):

a) Using loop.
b) Using recursion.

Problem 4*

Define a struct or class called FastFood that can store the following information: food name (string), food ID (string), cost (float), deliciousness (float).

- Write a function that receives an array of FastFood (as a parameter), print the name of all FastFoods with cost below 5.0 and deliciousness above 8.0.
- Write a function that receives an array of FastFood (as a parameter), return **two** FastFood instances with the highest deliciousness in that array.

Problem 5*

Write a **recursive** function to convert a number from binary to decimal. You must use the following prototype:

```
int bin2dec(int num)
```

Assume that the input of bin2dec is always valid binary format (consists of only 1 and 0). Example usage:

```
cout << bin2dec(1101); // Print 13
```