In this lab, we will learn how to allocate memory dynamically and de-allocate memory that is allocated dynamically. Recall the syntax for the operators **new** and **delete**.

```
Pointer = new   DataType;
Pointer = new   DataType  [IntExpression];

delete    Pointer;
delete   []   Pointer;

Example:
int *a = new int;
char *b = new char[5];
float *c = new float[3*someVariable+1];
double **d = new double*[10];
delete a;
delete [] b;
delete [] c;
delete [] d;
```

Before you begin: Create a C++ project. A project contains all the relevant files such as source codes, header files, and relevant resources. Also, under CodeBlocks, you can only debug your program under a project; single-file program debugging is not supported. Follow the steps below to create a project.

1. File ⇒ New ⇒ Project... ⇒ Console Application ⇒ Go.

2. The "Console Application" wizard appears:

   a. Next

   b. Select "C++" ⇒ Next.

   c. In "Project Title", enter the name you want to assign to the project (for example lab01). In "Folder to create project in", set to your working directory, (for example d:\labs). Accept the default for the rest ⇒ Next. A project directory "lab01" will be created under "d:\labs", with a project configuration filename of "lab01.cbp". You could later create more projects under this working directory "d:\labs".

   d. In "Compiler" field, accept the defaults of "GNU GCC Compiler" ⇒ Finish.

3. Under the "Management" pane ⇒ Choose "Projects" tab ⇒ Expand the project node "lab01" ⇒ Expand "Sources" node ⇒ Double-click "main.cpp", which is an example program to print "Hello, world!".

4. To build the program, select "Build" menu ⇒ Build.

5. To run the program, select "Build" menu ⇒ Run.

Tasks:

Task 1: Using the **new** operator, allocate an integer array of user specified size (the user gives the size of the array as input). Assign values to the array elements by taking user inputs and then print the values. Finally de-allocate the array using the **delete** operator.

Task 2: Using the **new** operator, allocate a two dimensional character array. Again the number of rows and columns are going to be provided by the user as input. All of the rows are the same size. Take character strings as input from the user and then print the strings. Finally de-allocate the array using the **delete** operator.