Lab 3. Methods

Write a class in C++ that has the following definition:

```
class Math
{
public:
 static int Add(int,int);
 static int Add(int,int,int);
 static int Add(double,double);
 static int Add(double,double,double);
 static int Mul(int,int);
 static int Mul(int,int,int);
 static int Mul(double,double);
 static int Mul(double,double,double);
 static int Add(int count,...); // sums up a list of integers
        static char* Add(const char *, const char *)
}
```

Organize the code in the following way:

- a header file called **Math.h**
- a cpp file called **Math.cpp** that contains the source code for class **Math**
- a main file called **main.cpp** that contains the main function and has an example on how to use **Math**. The example must include using all methods from the class.
- for the variadic method use pointers or **va_start** / **va_end** macros.
- Add(const char *, const char *) will allocate memory that can add the two existing string. If one of the strings is nullptr, this function will return nullptr;

Write a class in C++ that has the following definition:

```
class Canvas
{
 // add private data members
public:
 Canvas(int width,int height);
 void DrawCircle(int x, int y, int ray, char ch);
 void FillCircle(int x, int y, int ray, char ch);
 void DrawRect(int left, int top, int right, int bottom, char ch);
 void FillRect(int left, int top, int right, int bottom, char ch);
 void SetPoint(int x, int y, char ch);
 void DrawLine(int x1, int y1, int x2, int y2, char ch);
 void Print(); // shows what was printed
 void Clear(); // clears the canvas
}
```

Organize the code in the following way:

- a header file called **Canvas.h**
- a cpp file called **Canvas.cpp** that contains the source code for class **Canvas**
- a main file called **main.cpp** that contains the main function and has an example on how to use **Canvas**. The example must include using all methods from the class.
- use a matrix of type char for the canvas. A pixel is a cell in the matrix. A pixel is considered empty if it contains the space (value 32) character.

For the DrawLine algorithm use **Braseham** algorithm.