# Coding Convention

Main principle: Readability over briefness. (self-commenting code)

Methods should only do one thing and do it well. If possible 5 lines or less. If the method does multiple things divide it into multiple methods. If the method is too long think whether you can divide it into divide methods.

Classes should be only as long as much fits on a screen at once. If it’s too long it might be good to turn it into more classes.

Put empty lines between where it makes sense to divide the code for readability. For example, between handling input and handling logic.

Use comments above methods to say what it does, above classes to say what the class is, how to call its methods and what parameters objects made from this class need.

Use comments wherever they feel appropriate: divide in code, loops, change of what the code is doing at that place, also what do variables represent.

Camel notation for variables and functions:

int firstVariableName;

doSomething();

Classes start with capital letter:

class ClassName

{

...

};

Objects start with small letter:

ClassName objectOfClassName;

Curly brackets are always on their own line:

{

whatever is

between brackets…

}

Divide variables based on logic and not just their type + commenting variables:

int distanceA, distanceB; //variables for storing distance

int sizeA, sizeB; //variables for storing size

We can put all these ints into one line but it’s neater if they are logically separated and easier to find what you are looking for.

## **Example:**

void printArray(int \*array, int n)

{

   for (int i = 0; i < n; ++i)

     std::cout << array[i] << " " << std::flush;

   std::cout << std::endl;

}

void bubbleSort(int \*array, int n)

{

   bool swapped = true;

   int j = 0;

   int temp;

   while (swapped)

{

     swapped = false;

     j++;

     for (int i = 0; i < n - j; ++i)

{

       if (array[i] > array[i + 1])

{

         temp = array[i];

         array[i] = array[i + 1];

         array[i + 1] = temp;

         swapped = true;

       }

     }

   }

}

int main()

{

   int array[] = {94, 42, 50, 95, 333, 65, 54, 456, 1, 1234};

   int n = sizeof(array)/sizeof(array[0]);

   std::cout << "Before Bubble Sort :" << std::endl;

   printArray(array, n);

   bubbleSort(array, n);

   std::cout << "After Bubble Sort :" << std::endl;

   printArray(array, n);

   return (0);

}