Handout 3
C++ Programming
Deadline is
October 17

1a. Write a function **`count_chars`** that has the function head

```
void count_chars(char text[])
```

and that performs the following task:

`count_chars` prints a table listing the number of occurrences of the individual characters A to Z (capitalized as well as non-capitalized) in the argument **`text`**. The argument **`text`** is assumed to be a C-string that is terminated using a NULL-character.

Example:

The program:

```
int main() {
      char text[] = "Today is a nice day for having a little picnic.";
      count_chars(text);
      return 0;
} // end main
```

shall print:

```
A, a : 5
B, b : 0
C, c : 3
D, d : 2
E, e : 2
F, f : 1
G, g : 1
H, h : 1
I, i : 6
J, j : 0
K, k : 0
L, l : 2
M, m : 0
N, n : 3
O, o : 2
P, p : 1
Q, q : 0
R, r : 1
S, s : 1
T, t : 3
U, u : 0
V, v : 1
W, w : 0
X, x : 0
Y, y : 2
Z, z : 0
```

Answer:

```
#include <iostream>

void count_chars(char text[])
{
    static int frequency[26];
    if (text[0] == '\0')
    {
        for (size_t i = 0; i < 26; i++)
            printf("%c, %c : %d\n", char('A'+i), char('a'+i),frequency[i]);
        return;
    }
    int c = int(text[0]);
    if ('z' >= c && 'A' <= c)
    {
        frequency[(c >= 'a' ? c - 32 : c) -'A']++;
    }
    count_chars(text + 1);
}

int main()
{
    char text[] = "Today is a nice day for having a little picnic";
    count_chars(text);
    return 0;
}
```

1b. Modify/extend the function **`count_chars`** so that it additionally prints a bar diagram showing the occurrences of the characters A to Z. The height of each bar shall be equal to the number of occurrences of the corresponding character in the argument **`text`**.

For the above example, your output should look like:

```
          *
*         *
*         *
* *       *     *       *
* ***     *   * **      *      *
* *******   * *** *** *   *   *
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Answer:

```
#include <iostream>

void draw_plot(int frequency[])
{
    int max = *std::max_element(frequency, frequency + 26);
    for (int i = 0; i < max; i++) {
        for (int j = 0; j < 26; j++) {
            std::cout << (max - i <= frequency[j] ? '*' : ' ');
        }
        std::cout << std::endl;
    }
    for (int i = 'A'; i <= 'Z'; i++) {
        std::cout << char(i);
    }
    std::cout << std::endl;
}

void count_chars(char text[]) {
    static int frequency[26];
    if (text[0] == '\0') {
        draw_plot(frequency);
        return;
    }
```

```
    int c = int(text[0]);
    if ('z' >= c && 'A' <= c) {
        frequency[(c >= 'a' ? c - 32 : c) - 'A']++;
    }
    count_chars(text + 1);
}
```