ZIG ZAG CONVERSION

Presented
By
KAVINKISHORE E
125176042
MCA 1st B
PROBLEM SOLVING IN C PROGRAMMING

Course Coordinator : Aarthi Selvaraj

AIM

- To write a c program for ZIG ZAG CONVERSION using array, zig zag conversion in c programming typically involves rearranging characters in a string in a specific pattern.

ALGORITHM

- Create a function, let's call it zigzagConversion, that takes two parameters: a string (str) and the number of rows (numRows) for the Zigzag pattern.
- Calculate the length of the input string str using the strlen function and store it in a variable, let's call it len.
- Check if numRows is equal to 1 or greater than or equal to len. If either condition is true, there's no need to perform Zigzag conversion, so you can just print the original string and return.
- Create a character array of the same length as the input string to store the result. Initialize this array with null characters to ensure proper termination.

- Calculate the step size, which is equal to 2 * (numRows - 1).
- Initialize an index variable to keep track of the current position in the result array.
- Use a nested loop to iterate through each row from 0 to numRows - 1:

  a. Inside the outer loop, iterate through the characters of the input string with an inner loop. Start from the current row index and then add the step size in each iteration to reach the next character in the Zigzag pattern.

  b. Place the character at the current index of the result array and increment the index.

  c. If the current row is not the first or the last row (i.e., it's in the middle), check if there's a character in the original string that needs to be placed in the result array based on the Zigzag pattern. Calculate the index of this character using j + step - 2 * i. If such a character exists, place it in the result array and increment the index.

- After both loops have completed, the result array should contain the Zigzag converted string.
- Print the result array as the Zigzag converted string.
- In the main function, you can call the zigzagConversion function with your input string and the desired number of rows to perform the Zigzag conversion.

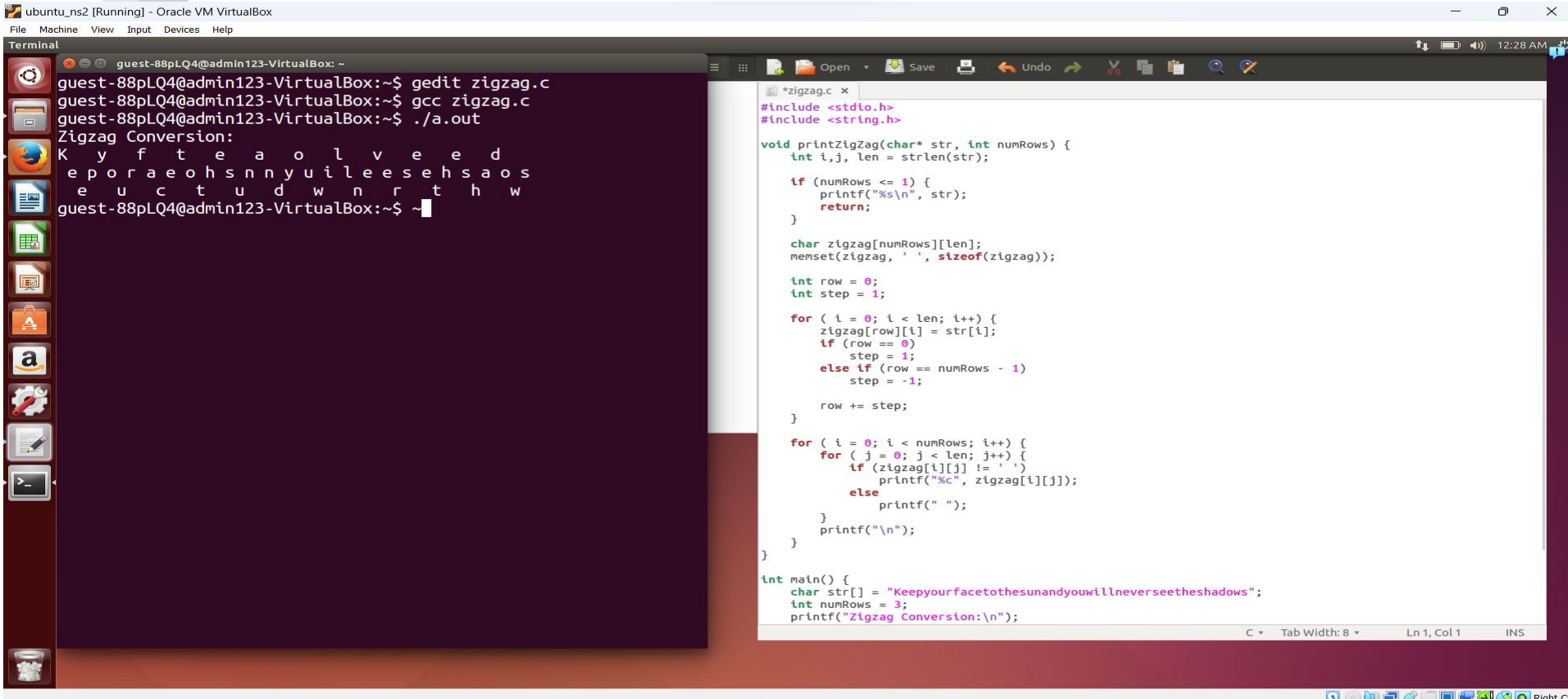
CODING

```
#include <stdio.h>
#include <string.h>
void convertToZigzag(char *str, int numRows) {
    if (numRows <= 1) {
        printf("%s\n", str);
        return;
    }
int len = strlen(str);
    char zigzag[len];
    int interval = 2 * (numRows - 1);
    int index = 0;
```

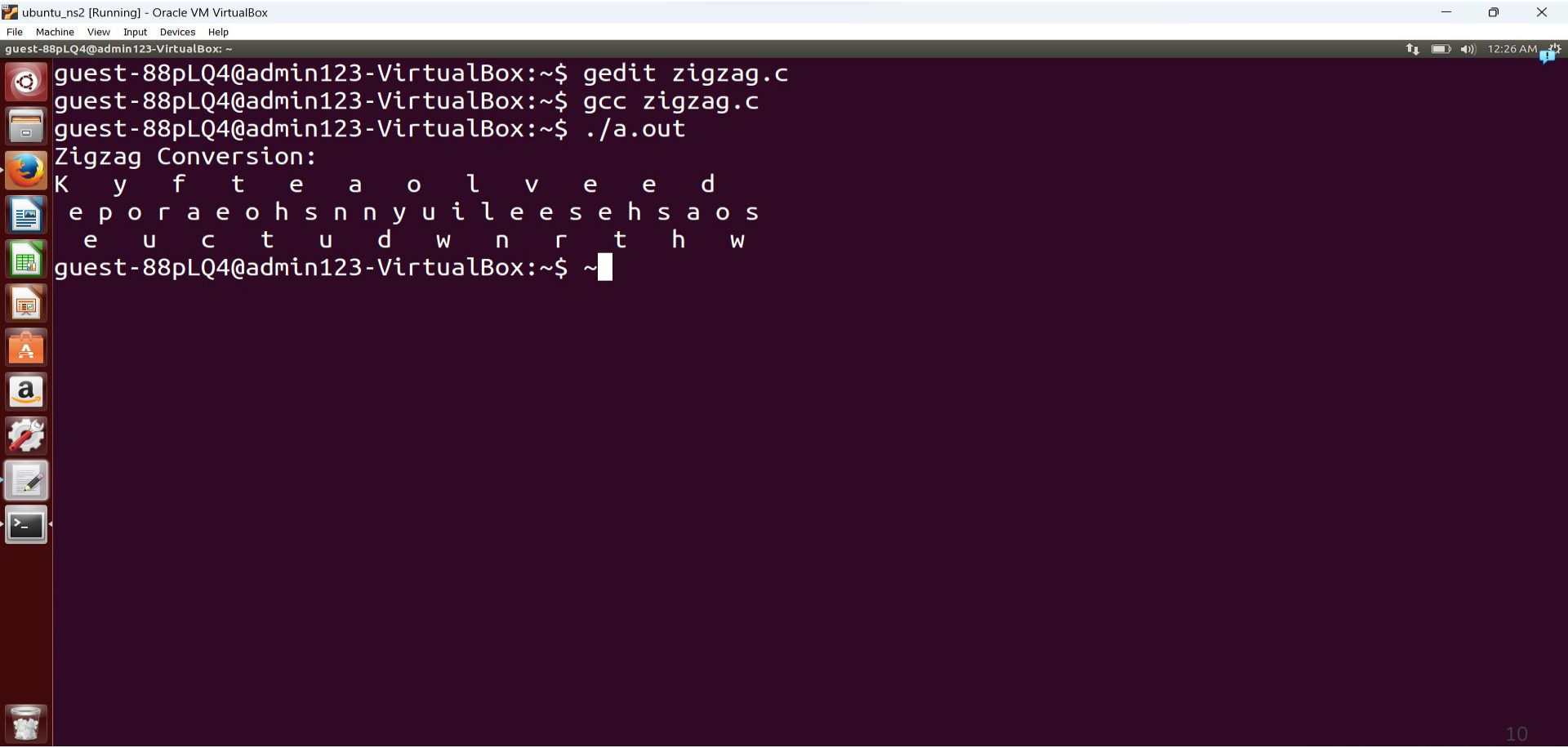
```
 for (int i = 0; i < numRows; i++) {
        int step = interval - 2 * i;
for (int j = i; j < len; j += interval) {
            zigzag[index] = str[j];
            Index++;
   if (step > 0 && step < interval && j + step < len) {
                zigzag[index] = str[j + step];
                index++;
            }
        }
    }
```

```
zigzag[index] = '\0';
    printf("%s\n", zigzag);
}
int main()
{
char str[] = "Keep your face to the sun and you will never see the shadows";
    int numRows = 3;
 printf("Original string: %s\n", str);
    printf("Zigzag converted string:\n");
    convertToZigzag(str, numRows);
   return 0;
}
```

SAMPLE OUTPUT



OUTPUT


ubuntu_ns2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
guest-88pLQ4@admin123-VirtualBox: ~
12:26 AM
guest-88pLQ4@admin123-VirtualBox:~$ gedit zigzag.c
guest-88pLQ4@admin123-VirtualBox:~$ gcc zigzag.c
guest-88pLQ4@admin123-VirtualBox:~$ ./a.out
Zigzag Conversion:
K y f t e a o l v e e d
e p o r a e o h s n n y u i l e e s e h s a o s
e u c t u d w n r t h w
guest-88pLQ4@admin123-VirtualBox:~$ ~

RESULT

- Thus the above c program is executed and verified successfully