

Online C Compiler - online edit: X

Convert JPG to PDF, Images JPG X

+

www.onlinegdb.com/online\_c\_compiler

Language C

Run Debug Stop Share Save Beauty

main.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 static void append_char(char **s, size_t *len, size_t *cap, char c) {
5     if (*len + 1 >= *cap) {
6         *cap = (*cap == 0) ? 64 : (*cap * 2);
7         *s = realloc(*s, *cap);
8         if (!*s) { perror("realloc"); exit(1); }
9     }
10    (*s)[(*len)++] = c;
11    (*s)[*len] = '\0';
12 }
13 char *bit_stuff(const char *input) {
14     if (!input) return NULL;
15     size_t cap = 0, len = 0;
16     char *out = NULL;
17     int count1 = 0;
18     for (size_t i = 0; input[i] != '\0'; ++i) {
19         char b = input[i];
20         if (b != '0' && b != '1') continue;
21         append_char(&out, &len, &cap, b);
22         if (b == '1') {
23             count1++;
24             if (count1 == 5) {
25                 append_char(&out, &len, &cap, '0');
26                 count1 = 0;
27             }
28         } else {
```

input

Original : 0111111011101111101111110  
Stuffed : 0111110101110111110011111010  
Unstuffed : 0111111011101111101111110

...Program finished with exit code 0  
Press ENTER to exit console.

About • FAQ • Blog • Terms of Use • Contact Us •  
GDB Tutorial • Credits • Privacy  
© 2016 - 2025 GDB Online

29°C  
Mostly sunny

Search

ENG IN 09:41  
27-09-2025

online compiler and debugger for c/c++

code. compile. run. debug. share.

IDE

## My Projects

Classroom **new**

**Learn Programming**

## Programming Questions

**Sign Up**

**Login**

[About](#) • [FAQ](#) • [Blog](#) • [Terms of Use](#) • [Contact Us](#) •

[GDB Tutorial](#) • [Credits](#) • [Privacy](#)

© 2016 - 2025 GDB Online

main.c

```

24         if (count1 == 5) {
25             append_char(&out, &len, &cap, '0');
26             count1 = 0;
27         }
28     } else {
29         count1 = 0;
30     }
31 }
32 return out;
33 }
34 char *bit_unstuff(const char *input) {
35     if (!input) return NULL;
36     size_t cap = 0, len = 0;
37     char *out = NULL;
38     int count1 = 0;
39
40     for (size_t i = 0; input[i] != '\0'; ++i) {
41         char b = input[i];
42         if (b != '0' && b != '1') continue;
43
44         if (b == '1') {
45             append_char(&out, &len, &cap, '1');
46             count1++;
47         } else {
48             if (count1 == 5) {
49                 // Skip stuffed 0
50                 count1 = 0;
51                 continue;

```

```
Original   : 011111101110111101111110
Stuffed    : 011111010111011110011111010
Unstuffed  : 011111101110111101111110
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

main.c

```

43-
44-     if (b == '1') {
45-         append_char(&out, &len, &cap, '1');
46-         count1++;
47-     } else {
48-         if (count1 == 5) {
49-             // Skip stuffed 0
50-             count1 = 0;
51-             continue;
52-         } else {
53-             append_char(&out, &len, &cap, '0');
54-             count1 = 0;
55-         }
56-     }
57- }
58- return out;
59- }
60- int main(void) {
61-     const char *input = "011111101110111110111110";
62-     char *stuffed = bit_stuff(input);
63-     char *unstuffed = bit_unstuff(stuffed);
64-     printf("Original   : %s\n", input);
65-     printf("Stuffed    : %s\n", stuffed);
66-     printf("Unstuffed  : %s\n", unstuffed);
67-     free(stuffed);
68-     free(unstuffed);
69-     return 0;
70- }

```

close ad [x]

```
Original : 0111111011011110111110
Stuffed  : 01111101011011111001111010
Unstuffed: 0111111011011110111110
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```