How can I print a string to the console at specific coordinates in C++?

Asked 14 years, 2 months ago Modified 5 years, 11 months ago Viewed 63k times

12

I'm trying to print characters in the console at specified coordinates. Up to now I have been using the very ugly `printf("\033[%d;%dH%s\n", 2, 2, "str");` But I just had to ask whether C++ had any other way of doing this. The problem is not even that it's ugly, the problem comes up when I try to make myself a prettier function like so:

```
void printToCoordinates(int x, int y, string text)
{
    printf("\033[%d;%dH%s\n", x, x, text);
}
```

It doesn't work, even if I typecast to `(char*)`. Another problem is that I have to print out the `\n` for the page to be refreshed... I just don't enjoy using `printf` in general.

Similarily to using `cout` instead of `printf`, I believe there should be a more recent way of doing this (ideally a way that allows me to easily write strings where I want on the screen, and ideally a way that doesn't required these weird symbols: `\033[%d;%dH`)

So, do any of you have what I'm looking for?

c++ printf coordinates cout

Share Edit Follow

asked Nov 3, 2009 at 23:46

Shawn
11.1k 18 81 129

1 What platform are you working on? – Jacob Nov 3, 2009 at 23:49

Ubuntu 9.04, I'm using g++ to compile – Shawn Nov 3, 2009 at 23:58

7 Answers

Sorted by: Highest score (default)

15

Curses is what you are looking for.

Share Edit Follow

answered Nov 3, 2009 at 23:48


Dima
38.9k 14 76 115

I can't find what to include to have access to curses. Nor can I find any documentation for the API..
– Shawn Nov 4, 2009 at 0:01


14

I remember using `gotoxy(x,y)` in Turbo C++ (conio.h) - don't know if it'll work for you though. It moves the cursor to the coordinates specified by `x` and `y`.

EDIT: If you're using Windows, here's a `gotoxy` clone:

```
#include <windows.h>

void gotoxy(int x, int y)
{
  COORD coord;
  coord.X = x;
  coord.Y = y;
  SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), coord);
}
```

Share Edit Follow

edited Sep 24, 2017 at 17:11

Ron
14.9k 4 34 48

answered Nov 3, 2009 at 23:51

Jacob
34.4k 14 111 166


7



What you are doing is using some very terminal specific magic characters in an otherwise pure C++ application. While this works, you will probably have a far easier time using a library which abstracts you from having to deal with terminal specific implementation details and provides functions that do what you need.

Investigate whether curses or ncurses libraries are available for your system.

Share Edit Follow

answered Nov 3, 2009 at 23:50

CB Bailey
765k 106 635 657


6

A few improvements to your function:

```
void printToCoordinates(int x, int y, const char *format, ...)
{
    va_list args;
```

```
    va_start(args, format);
    printf("\033[%d;%dH", x, y);
    vprintf(format, args);
    va_end(args);
    fflush(stdout);
}
```

This version:

- allows you to use any arbitrary format string and variable argument lists
- automatically flushes `stdout` without printing a newline
- uses `x` and `y` in the format string (your use of `x` and `x` may have been a typo)

However, because varargs is essentially a C feature and doesn't really understand C++ objects, you'd have to call it like this:

```
printToCoordinates(10, 10, "%s", text.c_str());
```

A better option really is to use curses (for Unix-like platforms) or Win32 console functions (for Windows) as mentioned in other answers.

Share Edit Follow

edited Oct 28, 2017 at 9:15

answered Nov 3, 2009 at 23:51

Greg Hewgill
963k 185 1155 1289

what does the fflush(stdout); part do? – Shawn Nov 3, 2009 at 23:59

Normally stdout is "buffered" which means the C runtime library queues up what you print and only sends it to the console when (a) you output a newline, (b) the queue fills up, or (c) you manually flush the file. Using `fflush()` in this case is more straightforward and obvious than printing a newline (and it will prevent scrolling problems if you're trying to print stuff on the bottom line of the screen). – Greg Hewgill Nov 4, 2009 at 0:32

4

First:

```
void printToCoordinates(int x, int y, string text)
{
    printf("\033[%d;%dH%s\n", x, x, text);
}
```

You don't want to copy the string argument, you want to pass it by (`const`) reference. Also, the (only) right way to get a `char*` from a `std::string` is to call its `c_str()` member function:

```
void printToCoordinates(int x, int y, const std::string& text)
{
```

```
    printf("\033[%d;%dH%s\n", x, x, text.c_str());
}
```

As to your question: C++ has no way to do what you want, but it allows you to use platform-specific ways to do it. You would have to tell us your platform in order to get meaningful answers.

Share Edit Follow

answered Nov 3, 2009 at 23:53

sbi
221k 46 258 444

Should be printf(...., x, y, text); not printf(...,x,x,text) .. cannot edit above. – mosh Jun 16, 2020 at 15:32

0

```
void screenpos(int x,int y,char textyowanawrite[20])
{
//printf for right shift
// \n for downward shift
//loops through the rows and shifts down
for(int row=0;row<=y;row++)
{
printf("\n");
for (int i = 0; i < x; i++)
{
printf("%s "," " );
}
}
printf("%s ",textyowanawrite );
}
```

//this should work to certain extinct only problem is u cant go from somewhere like 4,4 to 2,2 thats the problem

Share Edit Follow

answered Jan 17, 2018 at 4:18

shrinjoy biswas
9 1

-1

I have a little different method. Not sure whether this is better than ncurses package, so i leave that for the upvoters to decide.

You can use **Graphics** package in C++ to output a text to a specific coordinate on your working screen. The syntax is `outtextxy(x, y, text) ;` Where x & y are coordinates.

One example is:

```
int main(void) {

    int gdriver = DETECT, gmode;
```

```
    int x = 200, y = 200;


    initgraph(&gdriver, &gmode, "c:\\tc\\bgi");


    outtextxy(x, y, "Hello World");
    closegraph();
}
```

This little program will print **Hello World** in the coordinate (200,200).

For reference to what graphics package can do visit this link

Share Edit Follow

edited Jan 17, 2018 at 5:40

rimonmostafiz
1,371 1 16 33

answered Jan 17, 2018 at 4:42

Reaganrewop
1 1

The graphics package is non-standard, and probably unavailable on Ubuntu, the OP's plateforml – Basile Starynkevitch Jan 17, 2018 at 5:44