



EXERCISES — Date Format

version #7be580532266ed398481e31366afcc24b1950c2a



**The way is lit. The path is clear.
We require only the strength to follow it.**

Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2022-2023 Assistants <assistants@tickets.assistants.epita.fr>

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto someone else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Goal	3
2	Example	4

*<https://intra.assistants.epita.fr>

File Tree

```
date_format/  
├── date_format.c  (to submit)  
└── date_format.h  (to submit)
```

Authorized functions : You are only allowed to use the following functions

- time(2)
- gmtime(3)
- strdup(3)
- strtok_r(3)
- malloc(3)
- calloc(3)
- free(3)
- sprintf(3)

Authorized headers : You are only allowed to use the functions defined in the following headers

- err.h
- errno.h
- assert.h
- stddef.h

Compilation : Your code must compile with the following flags

- -std=c99 -pedantic -Werror -Wall -Wextra -Wvla

Main function : None

1 Goal

Implement the following function:

```
char *date_format(const char *format);
```

This function takes as input a string representing a date format. It will contain M, D and Y in any order and exactly once, separated by only one of the following separation characters: ; : , - _ . However the second delimiter might differ from the first one. The delimiters will not appear neither at the beginning nor the end of the string. No spaces will be present in the string. D and M should be on 2 characters and Y on 4 e.g. 01-01-2017.

For example:

```

// Valid formats.
char *format1 = "M-D-Y";
char *format2 = "D_M_Y";
char *format3 = "Y,D,M";
char *format4 = "M;Y;D";
char *format1 = "M-D_Y";

// Your function will never be called with those:
char *format5 = "B;C;E";
char *format6 = "Y;D;M;Y;D;M";
char *format7 = "Y;:D;_M";
char *format8 = ";Y;D;M;";
char *format9 = "Y , D , M";

```

If any error occurs, your function should return a `NULL` pointer. Errors include a missing separating character, a missing format (Y, D, M) or any common errors when using `malloc` etc.

The resulting string must be the current time, formatted as requested in the input format.

2 Example

```

char *output = date_format("Y,D,M");
printf("%s\n", output); // 2022,06,10
free(output);

output = date_format("M;Y:D");
printf("%s\n", output); // 10;2022:06
free(output);

output = date_format("Y;D"); // NULL
output = date_format("MYD"); // NULL

```

Be careful!

Your code will be compiled with `-D_XOPEN_SOURCE=700`. There should therefore be no errors when compiled this way.

The way is lit. The path is clear. We require only the strength to follow it.