B2 - Unix System Programming

B-PSU-200

tetris

80's puzzle game in terminal phase

2.4

tetris

binary name:	tetris
repository name:	PSU_tetris_$ACADEMICYEAR
language:	C
compilation:	via Makefile, including re, clean and fclean rules

- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.
- All the bonus files (including a potential specific Makefile) should be in a directory named *bonus*.
- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Authorized functions: *rand*, *srand*, *getopt*, *getopt_long*, *clock*, and all functions used for PSU projects until this point

The goal of this project is to recreate the Tetris game in a UNIX terminal, with the Gameboy version rules. You have to use ncurses. In the folder of your binary, there must be a *tetriminos* directory, which contains files that describe the game pieces.

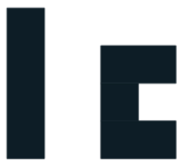
```
Terminal
~/B-PSU-200> ls ./tetriminos/
bar.tetrimino                square.tetrimino        5.tetrimino       7.tetrimino
inverted-L.tetrimino         4.tetrimino             6.tetrimino
```

These files are composed in the following way:

- on the first line, the size and color of the piece in this format: *width height color_code\n* (the number of the color corresponds to the ncurses capacity's color numbers),
- on the *h* following lines (where *h* is the height of the tetrimino), the piece's shape composed with asterisks (*) and spaces (' ').

For instance, these pieces correspond to the opposite files:

```
~/B-PSU-200> cat -e bar.tetrimino
1 4 2$
* $
* $
*$
* $
~/B-PSU-200> cat -e 6.tetrimino
2 3 6$
**$
*$
**$
```



The pieces (randomly chosen) fall from the top of the map and pile up on the bottom. Each time a line is completed, it disappears, leaving all of the pieces above it to fall.
The level increases by 1 for every 10 deleted lines. The falling speed increases proportionally to the level.
When it is no longer possible for pieces to fall from the top of the map, the player loses.

You can find the full gamerules here.

When the game begins, the terminal must delete all content. Then, it must show (at least):

- the main map,
- a preview of the next tetrimino to fall,
- current score, high score, current number of completed lines, level, timer.

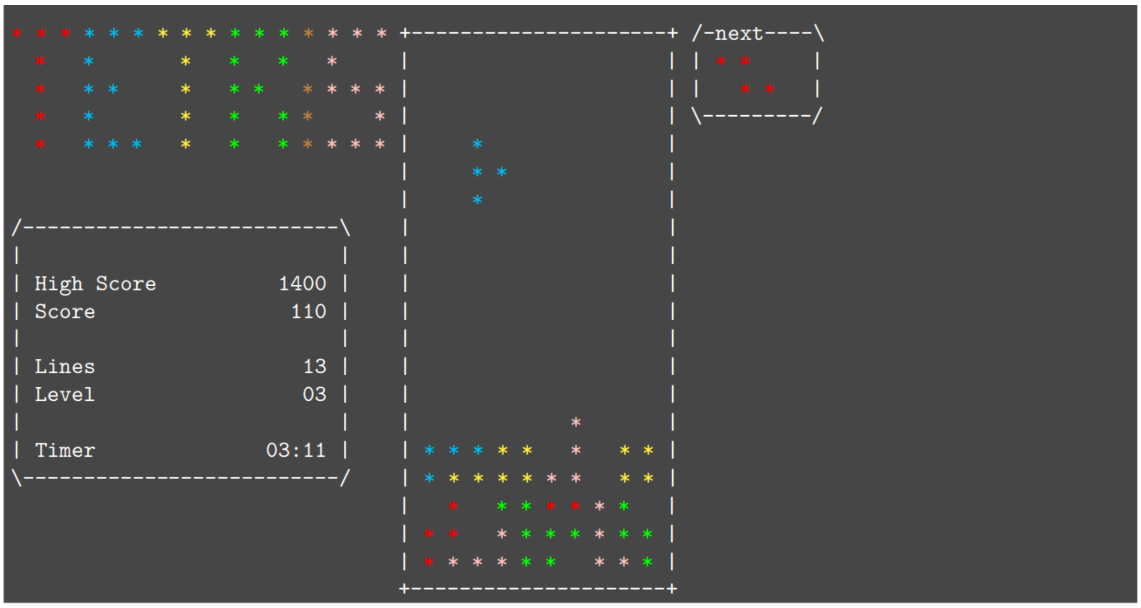
If the terminal is too small to host the map, the game doesn't start; an error message is printed, asking the user to enlarge his/her terminal.

For the gameplay, refer to the following usage:

```
Terminal
~/B-PSU-200> ./tetris --help | cat -e
Usage: ./tetris [options]$
Options:$
  --help               Display this help$
  -L --level={num}     Start Tetris at level num (def: 1)$
  -l --key-left={K}    Move the tetrimino LEFT using the K key (def: left arrow)$
  -r --key-right={K}   Move the tetrimino RIGHT using the K key (def: right arrow)$
  -t --key-turn={K}    TURN the tetrimino clockwise 90d using the K key (def: top a
  -d --key-drop={K}    DROP the tetrimino using the K key (def: down arrow)$
  -q --key-quit={K}    QUIT the game using the K key (def: 'q' key)$
  -p --key-pause={K}   PAUSE/RESTART the game using the K key (def: space bar)$
  --map-size={row,col} Set the numbers of rows and columns of the map (def: 20,10)$
  -w --without-next    Hide next tetrimino (def: false)$
  -D --debug           Debug mode (def: false)$
```

The K parameter of key binding options is the key code returned by `getch`/`wgetch` when `keypad` is active in decimal.

Here is a display example:



```
* * * * * * * * * * * * * * * * +---------------------+ /-next----\
  *   *       *   *   *   *     |                     | | * *     |
  *   * *     *   * *   * * * * |                     | |   * *   |
  *   *       *   *   * *     * |                     | \---------/
  *   * * *   *   *   * * * * * |     *               |
                                |     * *             |
                                |     *               |
/--------------------------\    |                     |
|                          |    |                     |
| High Score          1400 |    |                     |
| Score                110 |    |                     |
|                          |    |                     |
| Lines                 13 |    |                     |
| Level                 03 |    |                     |
|                          |    |             *       |
| Timer              03:11 |    | * * * * *   *   * * |
\--------------------------/    | * * * * * * *   * * |
                                |   *   * * * * * *   |
                                | * *   * * * * * * * |
                                | * * * * * *   * * * |
                                +---------------------+
```

Gameplay will be evaluated by your pedagogical team. Feel free to choose the graphic style you want, as long as its in ncurses and in a terminal.

Debug mode

Half of your project will be evaluated through the debug mode with automated tests, implement it scrupulously.

When in debug mode, display the following information on the standard output and exit:

```
Terminal
~/B-PSU-200> ./tetris -d 120 -D --key-turn=32 -p 112 | cat -e
Key left: KEY_LEFT (260)$
Key right: KEY_RIGHT (261)$
Key turn:   (32)$
Key drop: x (120)$
Key quit: q (113)$
Key pause: p (112)$
Next: Yes$
Level: 1$
Size: 20*10$
$
Number of tetriminos: 7$
Tetriminos '4': error$
Tetriminos '5': size 1*1, color 4$
*$
Tetriminos '6': size 2*3, color 6$
*$
**$
 *$
Tetriminos '7': size 5*4, color 3$
 * *$
* * *$
 * *$
  *$
Tetriminos 'bar': size 1*4, color 2$
*$
*$
*$
*$
Tetriminos 'inverted-L': size 2*3, color 5$
 *$
 *$
**$
Tetriminos 'square': size 2*2, color 1$
**$
**$
```

Your program must return 84 if there is not at least 1 valid tetrimino in `./tetriminos/`.

Bonus

Many nice bonus points are possible.
Unleash your imagination, and have fun:

- save the round to play again later,
- save High Scores with player names (hall of fame),
- High Score board,
- miscellaneous animations,
- music and sound effects,
- multi-player game (cf tetriNET for instance).