

Fruit reloaded code style guide

Preface

This document attempts to explain the basic styles and patterns that are used in the Fruit reloaded codebase. New code should try to conform to these standards, so that it is as easy to maintain as existing code. Of course every rule has an exception, but it's important to know the rules nonetheless!

Best practices

Abstract goals we strive for:

- simplicity
- readability
- bugfree (asserts)

When considering the goals above, each situation requires an examination of the circumstances and balancing of various trade-offs.

Please read the existing code base for a better understanding :)

General practices

- Use whole english words and short as possible
- Basicly we use **plain** C and only C++ if it brings a clear speed advantage
- Use **three** spaces for indentation (no tabs!)
- Use space between complex code lines

C/C++ practices

- Use NULL for pointers if needed
- Use // for comments
- Declare local variables at the top of each method (expectation speed up with C++)
- Declare **local** variables like 'this_is_a_local_variable'
- Declare (constants) **global** variables like „ThisIsAGlobalVariable“
- Don't initialize variables at the top and use „// init“ as comment
- Do not compare `x == true` or `x == false`. Use `(x)` or `(!x)` instead. `x == true`, in fact, is different from `if (x)!` (expectation asserts)
- Asserts using `x==false` boolean compare instead of `!x`
- Use ASSERT as much as possible
- Don't use if else statement without brackets
- Don't use else after return
- Use standard int as much as possible
- Use unsigned integer type if your variable represents a bit pattern (e.g. a hash value or a bitmask). (look at util.h)

File practies

This is an example:

```
/* Fruit reloaded, a UCI chess playing engine derived from Fruit 2.1
*
* Copyright (C) 2004-2005 Fabien Letouzey
* Copyright (C) 2012-2014 Daniel Mehrmann
* Copyright (C) 2013-2014 Ryan Benitez
*
* This program is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, either version 3 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program. If not, see <http://www.gnu.org/licenses/>.
*/
```

```
// fileName.cpp
```

```
// includes
```

```
#ifdef _WIN32
# include <windows.h>
#else
# include <cstring>
#endif
```

```
#include "board.h"
```

```
// constants
```

```
static const bool UseStats = false;
```

```
// types
```

```
enum gen_t {
    GEN_ERROR,
};
```

```
struct entry_t {
    uint32 lock;
};
```

```
// variables
```

```
static int PosLegalEvasion;
static int PosSEE;
```

```
// prototypes

static void      mutex_init      ();
static void      mutex_close    ();

static void      thread_create   ();
static void      thread_exit    ();


// functions

// move_eval()

void move_eval(const board_t * board, int eval) {

    int index;
    int best_value, value;
    int move;

    ASSERT(board!=NULL);
    ASSERT(value_is_ok(eval));
}

// end of xxx.cpp
```