

程設hw04 40947046S 劉佩昀

hw0401

1. ./hw0401
2. 輸入檔名實不要忘記輸副檔名。
3. 以下為程式介紹

在此題中，由於每個pixel要放大四倍，所以已我寫了四個function去填放大後pixel的值，

```
1 void black_1(pixel* a,pixel* b,pixel* c,pixel*d){
2     a->Blue=255;
3     a->Green=255;
4     a->Red=255;
5     b->Blue=0;
6     b->Green=0;
7     b->Red=0;
8     c->Blue=0;
9     c->Green=0;
10    c->Red=0;
11    d->Blue=255;
12    d->Green=255;
13    d->Red=255;
14 }
15
16 void black_2(pixel* a,pixel* b,pixel* c,pixel*d){
17     a->Blue=0;
18     a->Green=0;
19     a->Red=0;
20     b->Blue=255;
21     b->Green=255;
22     b->Red=255;
23     c->Blue=255;
24     c->Green=255;
25     c->Red=255;
26     d->Blue=0;
27     d->Green=0;
28     d->Red=0;
29 }
30
31 void white(pixel* a,pixel* b,pixel* c,pixel*d){
32     a->Blue=255;
33     a->Green=255;
34     a->Red=255;
35     b->Blue=0;
36     b->Green=0;
37     b->Red=0;
38     c->Blue=0;
39     c->Green=0;
40     c->Red=0;
41     d->Blue=255;
42     d->Green=255;
43     d->Red=255;
44 }
45
```

```

46 void gray(pixel* a,pixel* b,pixel* c,pixel* d,uint8_t color){
47     if(color>127){
48         a->Blue=255;
49         a->Green=255;
50         a->Red=255;
51         b->Blue=255;
52         b->Green=255;
53         b->Red=255;
54         c->Blue=255;
55         c->Green=255;
56         c->Red=255;
57         d->Blue=255;
58         d->Green=255;
59         d->Red=255;
60     }else{
61         a->Blue=0;
62         a->Green=0;
63         a->Red=0;
64         b->Blue=0;
65         b->Green=0;
66         b->Red=0;
67         c->Blue=0;
68         c->Green=0;
69         c->Red=0;
70         d->Blue=0;
71         d->Green=0;
72         d->Red=0;
73     }
74 }

```

hw0402

1. ./hw0402
2. 使用fprintf把文字寫入新的.h檔和.c檔

hw0403

1. ./hw0403
2. code一行不能超過1024，不然我的buffer會爆掉。
3. 檔名請放在最後一個argument
4. 以下為程式介紹：

使用getopt_long

```

1 struct option long_options[] = {
2     { "linenum" , 0, NULL, 'n' },
3     { "color" , 0, NULL, 'c' },
4     { "language", 1, NULL, 'l' },
5     { 0 , 0, 0 , 0 },
6 };

```

//規定：最後一定要有這行，表示結束，這是man的規定。

由於題目有兩種選擇：上色或加行數

所以我分成上色、加行數、上色且加行數，這三個function。

至於比對關鍵字的部分寫死在code裡，長這樣：

```

1  const char *rust_language[]={
2      "as"      , "break" , "const"   , "continue",
3      "crate"   , "else"  , "enum"    , "extern",
4      "false"   , "fn"    , "for"     , "if",
5      "impl"    , "in"    , "let"     , "loop",
6      "match"   , "mod"   , "move"    , "mut",
7      "pub"     , "ref"   , "return"  , "self",
8      "self"    , "static", "struct"  , "super",
9      "trait"   , "true"  , "type"    , "unsafe"
10     "use"     , "where" , "while"   , "async",
11     "await"   , "dyn"   , "abstract", "become",
12     "box"     , "do"    , "final"   , "macro",
13     "override", "priv"  , "typeof"  , "unsized",
14     "virtual" , "yeild" , "try"     , "union",
15     "'static"
16 };
17 const char *java_language[]={
18     "abstract", "assert"      , "boolean" , "break",
19     "byte"     , "case"        , "catch"   , "char",
20     "class"    , "const"       , "continue", "default",
21     "do"       , "double"      , "else"    , "enum",
22     "extends"  , "final"       , "finally" , "float",
23     "for"      , "if"          , "goto"    , "implements",
24     "import"   , "instanceof" , "int"     , "interface",
25     "long"     , "native"     , "new"     , "package",
26     "private"  , "protected"  , "public"  , "return",
27     "short"    , "static"     , "strictfp", "super",
28     "switch"   , "synchronized", "this"    , "throw",
29     "throws"   , "transient"  , "try"     , "void",
30     "volatile", "while"      , "_"
31 };
32
33 const char *js_language[]={
34     "await" , "break" , "case"      , "catch"      ,
35     "class" , "const" , "continue"  , "debugger"   ,
36     "default", "delete", "do"        , "else"       ,
37     "enum"   , "export", "extends"   , "false"      ,
38     "finally", "for"   , "function" , "if"         ,
39     "import" , "in"    , "instanceof", "new"        ,
40     "null"   , "return", "super"   , "switch"     ,
41     "this"   , "throw" , "true"    , "try"        ,
42     "typeof" , "var"   , "void"    , "while"      ,
43     "with"   , "yield"
44 };
45
46 const char *c_plus_language[]={
47     "alignas"      , "alignof"      , "asm"          , "auto"          ,
48     "bool"         , "break"        , "case"         , "catch"         ,
49     "char"         , "char16_t"     , "char32_t"     , "class"         ,
50     "const"        , "constexpr"    , "const_case"   , "continue"      ,
51     "decltype"     , "default"      , "delete"       , "do"            ,
52     "double"       , "dynamic_case" , "else"         , "enum"          ,
53     "explicit"     , "export"       , "extern"       , "false"         ,
54     "float"        , "for"          , "friend"       , "goto"          ,
55     "if"           , "inline"       , "int"          , "long"          ,
56     "mutable"      , "namespace"    , "new"          , "noexcept"      ,
57     "nullptr"      , "operator"     , "private"      , "protected"     ,
58     "public"       , "register"     , "reinterpret_cast", "return"        ,

```

```

59     "short"      , "signed"      , "sizeof"      , "static"      ,
60     "static_assert", "static_cast", "struct"       , "switch"      ,
61     "template"   , "this"       , "thread_local", "throw"       ,
62     "true"       , "try"        , "typedef"     , "typeid"     ,
63     "typename"   , "union"     , "unsigned"    , "using"      ,
64     "virtual"    , "void"      , "volatile"    , "wchar_t"    ,
65     "while"      , "and"       , "and_eq"     , "bitand"     ,
66     "bitor"      , "compl"     , "not"        , "not_eq"     ,
67     "or"         , "or_eq"     , "xor"        , "xor_eq"     ,
68 };
69
70 const char *C_language[]={
71     "auto"      , "break"     , "case"       , "char"       ,
72     "const"     , "continue"  , "default"    , "do"         ,
73     "double"    , "else"      , "enum"       , "extern"     ,
74     "float"     , "for"       , "goto"       , "if"         ,
75     "inline"    , "int"       , "long"       , "register"   ,
76     "restrict"  , "return"    , "short"      , "signed"     ,
77     "sizeof"    , "static"    , "struct"     , "switch"     ,
78     "typedef"   , "union"     , "unsigned"   , "void"       ,
79     "volatile"  , "while"     , "_Alignas"   , "_Alignof"   ,
80     "_Atomic"   , "_Bool"     , "_Complex"   , "_Generic"   ,
81     "_Imaginary", "_Noreturn" , "_Static_assert", "Thread_local"
82 };

```

hw0404

1. ./hw0404
2. 能做到修改attribute、金錢、健康狀況