

hw03

compile

make

hw0301

如果輸入空字元會全顯示

做法：用strcasestr去找target

hw0302

使用usleep達到srt的效果

hw0303

1. 只可使用色深為24的bmp圖片，如果不是24的話會報錯。
2. 在 **輸入檔名** 和 **輸出檔名** 時務必在最後加上 **.bmp**，否則會報錯。
3. 輸入座標時請使用 (x,y) 的格式，若有奇怪格式會報錯。
4. 輸入r時如果有奇怪格式的話，會報錯。

讀bmp圖片，把每個像素都讀到我自己寫的structure，如：

```
1  typedef struct tagRGBQUAD {
2      uint8_t Blue;
3      uint8_t Green;
4      uint8_t Red;
5  }__attribute__((packed)) BYTE;
```

接著去計算每個pixel到焦點的距離，如果大於r就把該pixel設成白色。

hw0304

1. 只可使用色深為24的bmp圖片，如果不是24的話會報錯。
2. 在 **輸入檔名** 和 **輸出檔名** 時務必在最後加上 **.bmp**，否則會報錯。

先讀標頭檔，把標頭檔讀進來之後，更改標頭檔，具體來說有：

1. header.bpp須改成16。
2. header.compression 須設成3，因為我們要加調色盤。
3. header.offset 須將原本的值加上調色盤占用的byte數。
4. header.bitmap_size 須將原本的值除以24乘以16，因為現在一個pixel只占用2byte。
5. header.size 須在上面的值都更新完後做，設成header.bitmap_size+header.offset。

改完之後把新的標頭檔資料丟到新圖片裡。

接著我們須把調色盤的資料也丟到新圖片裡，如：

```
1 typedef struct _plate{
2     uint32_t    r;
3     uint32_t    g;
4     uint32_t    b;
5     uint32_t    a;
6 }__attribute__((packed)) plate;
```

```
1 const plate mask=
  {.r=0b1111100000000000, .g=0b0000011111100000, .b=0b0000000000011111, .a=0b000000
  000000000000};
2 fwrite(&mask, sizeof(mask), 1, output_bmp);
```

接著讀每個像素的資料，把每個像素都讀到我自己寫的二維陣列，此二維陣列的型別如：

```
1 typedef struct tagRGBQUAD {
2     uint8_t Blue;
3     uint8_t Green;
4     uint8_t Red;
5 }__attribute__((packed)) BYTE;
```

如此一來我就能得到所有pixel的資料，去遍歷所有pixel，然後使用bitwise算rgb的資料，存到對應的uint16_t陣列裡。