2022 Digital IC Design
Final Project

NAME	劉品宏
Student ID	P76101699

Functional Simulation Result of LZ77 Encoder

Testing Pattern 0	Pass	Testing Pattern 1	Pass	Testing Pattern 2	Pass

```
# cycle 21820, expect(1b,02,9) , get(1b,02,9) >> Pass
# cycle 2183f, expect(00,00,1) , get(00,00,1) >> Pass
# cycle 2185e, expect(00,00,$) , get(00,00,$) >> Pass
# ---------------------------------------------
# --------- Encoding finished, ALL PASS ---------
.
# cycle 1cdc6, expect(00,00,7) , get(00,00,7) >> Pass
# cycle 1cde7, expect(01,01,e) , get(01,01,e) >> Pass
# cycle 1ce15, expect(15,02,$) , get(15,02,$) >> Pass
# ---------------------------------------------
# --------- Encoding finished, ALL PASS ---------
.
# cycle 1e377, expect(19,02,7) , get(19,02,7) >> Pass
# cycle 1e39a, expect(0b,02,0) , get(0b,02,0) >> Pass
# cycle 1e3b9, expect(00,00,$) , get(00,00,$) >> Pass
# ---------------------------------------------
# --------- Encoding finished, ALL PASS ---------
# ---------------------------------------------
```

Functional Simulation Result of LZ77 Decoder

Testing Pattern 0	Pass	Testing Pattern 1	Pass	Testing Pattern 2	Pass

```
# cycle 02003, expect 9, get 9 >> Pass
#   == Decoding string "1"
# cycle 02004, expect 1, get 1 >> Pass
# ---------------------------------------------
# -------------- Decoding finished, ALL PASS
# ---------------------------------------------
# cycle 02003, expect 5, get 5 >> Pass
# cycle 02004, expect f, get f >> Pass
# ---------------------------------------------
# -------------- Decoding finished, ALL PASS
# ---------------------------------------------
cycle 02002, expect 2, get 2 >> Pass
cycle 02003, expect 7, get 7 >> Pass
cycle 02004, expect 0, get 0 >> Pass
---------------------------------------------
-------------- Decoding finished, ALL PASS
---------------------------------------------
```

Quality of Interpolated Results

Testing Pattern 0	PSNR: 23.81	Testing Pattern 1	PSNR: 24.51	Testing Pattern 2	PSNR: 27.91

Description of your design

我的演算法是基於 hw4 去做改進的，分成兩個部分，首先，先完成 hw4 的演算法，這時已經有 128*63 大小的圖片了，然後，我再將偶數行的 Pixel 重新再做一次插值演算法，這裡跟 hw4 稍有不同，如果要重新寫入某個 Pixel 值，我還多考慮了左右兩邊的 pixel 值，如果發生相減一樣的情形，優先權則是左右>上下>左上右下>右上左下。

Scoring = Pattern 0 PSNR + Pattern 1 PSNR + Pattern 2 PSNR
The higher, the better.