

國立成功大學			學年度第		學期第		次平時考試試卷		
評閱成績	教師簽章	學號	姓名	院系	學號	姓名	科目	名稱	開設班別
		E94971344	呂煥璇	工學院工科	系三年級101班			Computer Organization	工科系三年級101班

1. (A) 32 bits
 (B) 8 bits
 (C) 64 bits
 (D) 2^{30}
 (E) ISA (Instruction Set Architecture) 指令集架構
 (F) Input, Output, Control, Datapath, Memory

2. (A) False
 (B) False
 (C) True
 (D) False
 (E) True
 (F) True

3. R-format: op rs rt rd shamt funct
 bits \Rightarrow 6 5 5 5 5 6
 I-format: op rs rt Addr/immediate
 bits \Rightarrow 6 5 5 16
 J-format: op address
 6 26

4. die area = 0.4 cm^2
 defects per $\text{cm}^2 = 4$

$$\Rightarrow \text{Yield} = \frac{1}{(1 + 4 \times \frac{0.4}{2})^2} = 30.86\%$$

if defects per area 減少一半，
 則

$$\text{Yield}' = \frac{1}{(1 + 2 \times \frac{0.4}{2})^2} = 51.02\%$$

(A) yuck:

li \$t0, 1 // i = 1
 li \$t1, 1 // j = 1

loop:
 blt \$a0, 2, exit //
 add \$t2, \$t0, \$t1 // k = i + j
 move \$t0, \$t1 // i = j
 move \$t1, \$t2 // j = k
 addi \$a0, \$a0, -1 // n = n - 1
 j loop // jump to loop

exit:

move \$v0, \$t1 // return value = j
 jr \$ra // jump-register \$ra

(B) return value 5 - 4 (續寫轉背頁)

C = int yuck(int n)

{ int i; int j; i = 1; j = 1;

while (n >= 2) {

int k = i + j;

i = j;

j = k;

n = n - 1;

return i;

$$6. (A) -284_{10} = -2^{31} + 2^{30} + 2^{29} + 2^{28} + 2^{27} + 2^{26} + 2^{25} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{20} + 2^{19} + 2^{18} + 2^{17} + 2^{16} + 2^{15} + 2^{12} + 2^8 + 2^2 + 2^1$$

two's complement:

1111 1111 1111 1111 1001 0001 0000 0110

$$(B) 0x CA5D = CA5D_{hex}$$

$$= 1100 1010 0101 0000_2$$

$$(C) 0x FFF7 = FFF7_{hex}$$

$$= 1111 1111 1111 0111_2$$

$$= -2^{15} + 2^{14} + \dots + 2^4 + 2^2 + 2^1 + 2^0$$

$$= -9_{10}$$

$$(D) 0x 001F = 001F_{hex}$$

$$= 0000 0000 0001 1111_2 = 3$$

$$0x FFF7 = 1111 1111 1111 0111_2 = -9$$

$$0x FFF7 - 0x 001F = -9 - 3 = -12_{10}$$

$$= 1111 1111 1101 1000 \text{ (2's complement)}$$

$$= FFD8_{hex}$$

7. (A) addi \$19, \$0, 0x20 \Rightarrow 把 0x20 从 memory load to register, 且 \$19 = \$0

lw \$17, 0x04(\$19) \Rightarrow 把 Memory [1] 从 Memory load to \$17, base address 为 \$19

add \$20, \$19, \$16 \Rightarrow \$20 = \$19 + \$16

sw \$20, 0x08(\$19) \Rightarrow 把 \$20 save to Memory [2]

(B) addi \$1, \$0, 0x20 \Rightarrow 把 0x20 从 memory load to register, base address 为 \$19

lw \$2, 0x04(\$3) \Rightarrow 把 Memory [1] 从 Memory load to register, 且 \$1 = \$0 + 0x20

add \$0, \$3, \$1 \Rightarrow \$0 = \$3 + \$1, base address 为 \$3

bne \$0, \$1, loop \Rightarrow if \$0 \neq \$1, 则 branch to loop

(C) Yes \Rightarrow if \$0 \neq \$1, 则 branch to loop -2

8. bne \$t0, \$zero, Loop (若 \$t0 不为 0 则 branch)

-2