

# 問題用紙

2025	科目名	情報基礎（計算機システム）	1 / 3	通し番号	
------	-----	---------------	-------	------	--

問題 1 次の(1)～(6)に答えよ。

- (1) 二進数で表された数値 11010b を十進数で表せ。
- (2) 8 ビットの 2 の補数表現で表される整数の範囲を十進数で答えよ。
- (3) 5000 通りの記号を二進数を用いて表現したい。全ての記号に  $n$  ビットの同じ長さの二進数を用いるとする。この表現を実現する最小の整数  $n$  を十進数で答えよ。
- (4) 十進数で表された数値 3.65 を、以下の仕様の浮動小数点数表現により表現する。丸めを切り捨てで行うとき、その指数部と仮数部を答えよ。また、その誤差の絶対値を十進数で答えよ。

浮動小数点数表現の仕様：

符号部 …… 1 ビット（符号ありを 1、符号なしを 0）

指数部 …… 3 ビットの 2 の補数表現

仮数部 …… 4 ビット（けち表現を用い、整数部の 1 を記憶しない）

- (5) 上記の(4)で生じる誤差の名称を答えよ。
- (6) 図 1 の論理回路の真理値表を完成させよ。

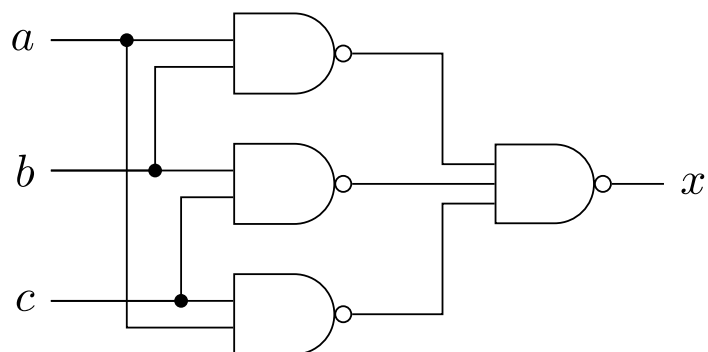


図 1：論理回路

# 問題用紙

2025	科目名	情報基礎（計算機システム）	2 / 3	通し番号	
------	-----	---------------	-------	------	--

**問題 2** 次の 32 ビットの MIPS のアセンブリ言語のプログラムについて、以下の間に答えよ。  
次のページに MIPS の命令表を表 1 に、レジスタ表を表 2 に示している。

```

1.          addi    $s0, $zero, 5
2.          addi    $s1, $zero, 3
3.      LOOP: add    $s1, $s1, $s1
4.          _____ # $s0 ← $s0 + (-1)
5.          bne     $s0, $zero, LOOP
6.          sw      $s1, -8($fp)

```

- (1) 1行目のニーモニックを機械語の命令のコードに変換せよ。
- (2) 6行目のニーモニックを機械語の命令のコードに変換せよ。
- (3) 1行目の下線部のオペランドのアドレッシングモードを、以下の(a)から(e)までの選択肢の中から一つ選べ。

(a) レジスタアドレッシング   (b) ベースアドレッシング   (c) 即値アドレッシング  
(d) PC 相対アドレッシング   (e) 擬似直接アドレッシング

- (4) MIPS アーキテクチャを採用したCPUにおける「add \$t0, \$s0, \$s1」の計算機の実行過程を、以下のステップを並べ替えて示せ。ただし、パイプライン処理は考慮する必要はない。順番はa, b, c等の記号で解答すること。
  - a. ALUの出力をレジスタ\$t0に格納する。
  - b. 命令をフェッチする。
  - c. レジスタ\$s0と\$s1の値をALUで加算する。
  - d. 命令をデコードし、レジスタ\$s0と\$s1の値を読み出す。
- (5) 6行目の命令実行時に、フレームポインタ\$fpの値が十六進数で0x10010010であったとする。このときの下線部の実効アドレスは何か。十六進数で答えよ。
- (6) 4行目の命令として、レジスタ\$s0に定数-1を加え、結果をレジスタ\$s0に格納する（即ち、レジスタ\$s0の値を1 減ずる）命令のニーモニックを示せ。表1の命令表のニーモニックを用いて答えよ。
- (7) 5行目の命令は何回実行されるか。
- (8) 6行目の実行によってメモリに格納される値を十進数で答えよ。
- (9) このプログラムは何をするプログラムか。即ち、1行目でレジスタ\$s0に設定された1以上の値（ $m$ とする）と2行目でレジスタ\$s1に設定された1以上の値（ $n$ とする）に対して、6行目でどのような値がメモリに格納されるか  $m$ と  $n$ を用いて答えよ。

# 問題用紙

2025	科目名	情報基礎（計算機システム）	3 / 3	通し番号	
------	-----	---------------	-------	------	--

表 1：MIPS 命令表

ニーモニック			動作	コード			
addi	\$rd,	\$rs, C	\$rd ← \$rs + C	001000ss	sssddddd	cccccccc	cccccccc
add	\$rd,	\$rs, \$rt	\$rd ← \$rs + \$rt	000000ss	sssttttt	dddddd000	00100000
sw	\$rs,	C(\$rb)	\$rs → M[\$rb + C]	101011bb	bbbsssss	cccccccc	cccccccc
bne	\$rs,	\$rt, C	\$rs ≠ \$rt なら C の行へ移動	000101ss	sssttttt	cccccccc	cccccccc

(注釈) コード中の sssss, ttttt, ddddd, bbbbbb は、それぞれニーモニック中の \$rs, \$rt, \$rd, \$rb のレジスタ番号 (表 2 参照)

コード中の cccc...ccc は、ニーモニック中の即値等の定数 C の 2 進表記 (負の数の場合は 2 の補数表現)

表 2：レジスタ表

名称	番号 (2 進)	用途
\$zero	00000	常に 0
\$s0 ~ \$s7	10000 ~ 10111	一時変数 (セーブされることを想定)
\$fp	11110	フレームポインタ