

電気情報工学科「コンピュータアーキテクチャ I」 中間試験

(2006年6月9日実施)

- A 課程担当 安浦 寛人
 B 課程担当 井上 弘士 (出題責任者)
 C 課程担当 中西 恒夫

【問題1】下の文章の括弧に適当な数値または記号を埋めて文を完成せよ。(40点:各4点)

- (1) 整数表現には、符号なし表現、符号つき絶対値表現、2の補数表現などがある。8ビットのデータ 11010101 は、符号なし表現としてみれば 10 進数表現で [ア], 符号つき絶対値表現としてみれば 10 進数表現で [イ], 2の補数表現としてみれば 10 進数表現で [ウ] を表している。
- (2) 8ビットの2の補数表現で表される最大の整数は、10 進数表現で [エ] であり、最小の整数は [オ] である。00000000 は、10 進数表現の [カ] を表し、11111111 は [キ] を表す。一般に、8ビットの2の補数表現 $a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0$ は、 $a_7 \times [ク] + a_6 \times 2^6 + a_5 \times 2^5 + a_4 \times 2^4 + a_3 \times 2^3 + a_2 \times 2^2 + a_1 \times 2^1 + [ケ]$ を表す。
- (3) 10 進表現で表された整数「-102」を8ビットの2の補数表現で表すと [コ] となる。

【問題2】あるプログラムAをプロセッサPで実行した場合、実行時間は100秒であった。また、そのうち20%が乗算命令の実行に、50%がメモリアクセス命令の実行に、30%がこれら以外の命令の実行に費やされた。ここで、プロセッサPにおいて、乗算命令ならびにメモリアクセス命令の実行能力を改善する場合を考える。ただし、これらの改善による他命令実行への悪影響は全く無いと仮定する。以下の問に答えよ。なお、必要であれば四捨五入して小数点以下第一位まで求めること。(20点:各5点)

- (1) 乗算命令の実行速度だけを従来の2倍に改善した。プログラムAの実行における速度向上比を求めよ。
- (2) メモリアクセス命令の実行速度だけを従来の5倍に改善した。プログラムAの実行における速度向上比を求めよ。
- (3) 上記(1)(2)の改善を両方施した。プログラムAの実行における速度向上比を求めよ。
- (4) 別の異なるプログラムBを改善前のプロセッサPで実行した場合、実行時間が160秒であった。また、その実行時間のうち、10%が乗算命令の実行に、15%がメモリアクセス命令の実行に、75%がこれら以外の命令の実行に費やされた。上記、(1)と(2)の改善を施した場合、プログラムBの実行における速度向上比はいくらになるか求めよ。

(参考)

$$\text{速度向上比} = \frac{\text{改善後の性能}}{\text{改善前の性能}} = \frac{\text{改善前の実行時間}}{\text{改善後の実行時間}}$$

$$\text{改善後の実行時間} = \left(\frac{\text{改善の影響を受ける実行時間}}{\text{改善度}} + \text{改善の影響を受けない実行時間} \right)$$

【問題3】下記のCプログラムおよびそれと同じ動きをするMIPSのアセンブラプログラムについて以下の問に答えよ。(40点)

Cプログラム:

```
if (i <= j){
    f = g + h + A[3];
    g = 0;
} else {
    f = [ア];
    i = i + j;
}
A[i] = f + [イ];
h = g + i;
```

アセンブリ言語プログラム:

```
slt    $t0, [ウ], [エ]
bne    [オ], $zero, ELSE
add    $s0, $s1, $s2
lw     $t0, 12([カ])
add    $s0, $s0, [キ]
add    $s1, $zero, $zero
j      L1
ELSE:  add    $s0, $zero, $s4
add    $s3, $s3, $s4
L1:    add    $t1, $s0, $s3
add    $t2, [ク], $s3
add    $t2, $t2, [ケ]
add    $t2, $t2, $s5
sw     [コ], 0($t2)
add    $s2, $s1, $s3
```

但し、変数のレジスタへの割り付けは下記の通りとする。また、メモリ番地に関してはバイト単位のビッグエンディアン方式を採用しており、全てのデータは整列化されている（全てのワード・データは4の倍数のアドレスが先頭となるよう配置されている）とする。さらに、「\$zero」は常にゼロを保持している特別なレジスタ、「\$t0～\$t7」は中間結果を一時的に保持する汎用レジスタとする。

| 変数 | レジスタ | 変数 | レジスタ |
|----|------|-----------|------|
| f | \$s0 | g | \$s1 |
| h | \$s2 | i | \$s3 |
| j | \$s4 | 配列 A 先頭番地 | \$s5 |

(1) 上記の C プログラムとアセンブリ言語プログラムが同じ動きをするように、(ア)～(コ)を埋めよ。(各3点)

(2) 各レジスタ\$s0, \$s1, \$s2, \$s3, \$s4, \$s5の初期値をそれぞれ10進数で20, 3, 65, 5, 7, 100とする。このプログラムの実行後の各レジスタの値を示せ。なお、配列Aは全ての要素に関して初期値を1(10進数)とする。(全て正解で10点)

参考資料 (MIPSの命令:「\$O」はレジスタを表す)

- **add \$O, \$Δ, \$□**
 - 符号つき整数(2の補数表現)の加算を行う($\$Δ + \$□ \rightarrow \$O$)。
- **slt \$O, \$Δ, \$□**
 - 第二オペランドと第三オペランドを2の補数表現された符号つき整数とみなして、比較結果を\$Oに書き込む($\$Δ < \$□ \rightarrow \O)。
 - 比較結果は、比較条件が成り立つとき1、そうでないとき0である。
- **lw \$O, n(\$Δ)**
 - 1ワード・データをメモリからレジスタへ転送($[\$Δ + n]$ 番地 \rightarrow \$O)。
 - オフセットnには、16bitの2の補数表現の整数(-32768~32767)を指定。
 - 対象番地は4の倍数でなければならない。
- **sw \$O, n(\$Δ)**
 - 1ワード・データをレジスタからメモリへ転送($\$O \rightarrow$ $[\$Δ + n]$ 番地)。
 - オフセットnには、16bitの2の補数表現の整数(-32768~32767)を指定。
 - 対象番地は4の倍数でなければならない。
- **bne \$Δ, \$□, Label**
 - \$Δと\$□が等しくなければ「Label」へ分岐。