

問 12 次のアセンブラプログラムの説明及びプログラムを読んで、設問1, 2に答えよ。

〔プログラムの説明〕

浮動小数点数の加算を行う副プログラム FADD である。

(1) 浮動小数点数は、メモリ中の連続する2語に次の形式で格納する。

メモリ番地	α		$\alpha + 1$
S	未使用	E	M
(1)	(7)	(8)	(16)

S : 符号部 (正は0, 負は1) () 内はビット数
 未使用 (ビット0を詰める)
 E : 指数部 (0~255の符号なし2進整数, 実際の指数に127を加算した値)
 M : 仮数部 (2進固定小数点数, 最上位ビットの左に小数点があると仮定する)

この形式で表される浮動小数点数は $(-1)^S \times 0.M \times 2^{E-127}$ である。

S=E=M=0でゼロを表現する (M=0のときは, 必ずS=E=0であること)。

(2) FADD は、被加数と加数について、いずれか指数の大きい方にもう一方の指数をそろえてから仮数を加算し、結果がゼロの場合を除き正規化 (仮数部の最上位ビットが1となるように指数部を調整) する。指数部調整の過程で、アンダフロー (指数部が0未満になるので正規化が不可能)、オーバフロー (指数部が255を超えるので正規化が不可能) は発生しないものとする。

正規化の例

[E = 10000100 M = 0010000000000000]
 ↓ 正規化 (M を 2 ビット左に論理シフトし, E から 2 を減じる)
 [E = 10000010 M = 1000000000000000]

(3) FADD は、被加数 X, 加数 Y 及び結果 Z の格納領域の先頭番地を、それぞれ GR1, GR2, GR3 に設定して呼び出される。

(4) FADD から戻るとき、汎用レジスタ GR1 ~ GR7 の内容は元に戻す。

[プログラム]

(行番号)

```

1  FADD    START                ; Z ← X + Y
2  RPUKSH
3  PUSH   0,GR3                ; 結果 Z の格納領域の先頭番地を退避
4  LD     GR4,0,GR1
5  AND    GR4,#00FF            ; Ex: X の指数
6  LD     GR5,0,GR2
7  AND    GR5,#00FF            ; Ey: Y の指数
8  LD     GR6,1,GR1            ; Mx: X の仮数
9  LD     GR7,1,GR2            ; My: Y の仮数
10 ; 加算前の準備 (指数をそろえる)
11 ; GR4 ← max(Ex,Ey) , GR6 ← 調整済 Mx , GR7 ← 調整済 My
12 LD     GR3,GR4
13 CPL   GR4,GR5
14 JZE   MADD                  ; Ex = Ey の場合
15 JMI   BIGEY                  ; Ex < Ey の場合
16 SUBL  GR3,GR5
17 SRL   GR7,0,GR3            ; My を調整
18 JUMP  MADD
19 BIGEY  a
20 SUBL  GR5,GR3
21 SRL   GR6,0,GR5            ; Mx を調整
22 ; 符号を考慮した仮数の加算
23 ; Sz: Z の符号 , Ez: Z の指数 , Mz: Z の仮数
24 ; GR4 ← (Sz,Ez) , GR5 ← Mz
25 MADD  LD   GR1,0,GR1        ; X の符号の検査
26 JMI   XMINUS                ; 負の場合
27 LD    GR2,0,GR2            ; Y の符号の検査
28 b
29 LD    GR5,GR6                ; X ≥ 0, Y ≤ 0 の場合
30 SUBL  GR5,GR7                ; Mz ← 調整済 Mx - 調整済 My
31 JUMP  SCHECK
32 XMINUS LD  GR2,0,GR2
33 JMI   YMINUS
34 LD    GR5,GR7
35 SUBL  GR5,GR6
36 c
37 YMINUS OR  GR4,#8000        ; Sz に負符号を設定
38 ADDMX LD  GR5,GR6
39 ADDL  GR5,GR7                ; Mz ← 調整済 Mx + 調整済 My
40 JOV  ADJST                ; けた上がりがある場合の正規化
41 JUMP  NORM

```

```

42 SCHECK JOV  NEGMZ      ; Mz の符号を検査
43          JUMP  NORM
44 NEGMZ   OR   GR4,=#8000 ; Sz に負符号を設定
45          XOR  GR5,=#FFFF ; Mz ← -Mz
46          ADDL GR5,=1
47 ; 加算結果の正規化
48 NORM    LD   GR5,GR5    ; ゼロチェック
49          JNZ  LOOP
50          LD   GR4,=0
51          JUMP FIN
52 LOOP    LD   GR5,GR5    ; 正規化完了?
53          JMI  FIN
54          
55          SUBL GR4,=1
56          JUMP LOOP
57 ADJST   SRL  GR5,1
58          OR   GR5,=#8000 ; Mz の最上位ビットを 1 に設定
59          
60 FIN     POP  GR3
61          ST   GR4,0,GR3 ; 結果 Z の格納
62          ST   GR5,1,GR3
63          RPOP
64          RET
65          END

```

設問 1 プログラム中の に入れる正しい答えを、解答群の中から選べ。

aに関する解答群

- | | | | | | |
|---|------|-----------|---|------|-----------|
| ア | ADDL | GR4,GR5 | イ | ADDL | GR5,GR4 |
| ウ | LD | GR4,GR5 | エ | LD | GR5,GR4 |
| オ | SRL | GR4,0,GR5 | カ | SRL | GR5,0,GR4 |

bに関する解答群

- | | | | | | |
|---|-----|--------|---|-----|--------|
| ア | JMI | ADDMXY | イ | JMI | XMINUS |
| ウ | JMI | YMINUS | エ | JPL | ADDMXY |
| オ | JPL | XMINUS | カ | JPL | YMINUS |

cに関する解答群

ア JOV ADDMXY
ウ JPL ADDMXY
オ JUMP ADDMXY

イ JOV SCHECK
エ JPL SCHECK
カ JUMP SCHECK

dに関する解答群

ア SLL GR5,0,GR4
ウ SRA GR5,0,GR4
オ SRL GR5,0,GR4

イ SLL GR5,1
エ SRA GR5,1
カ SRL GR5,1

eに関する解答群

ア ADDL GR4,=1
ウ SLL GR4,1
オ SUBL GR4,=1

イ ADDL GR5,=1
エ SLL GR5,1
カ SUBL GR5,=1

設問2 プログラムを減算用に変更する場合、行番号27及び32の直後に追加する命令として正しい答えを、解答群の中から選べ。

なお、プログラム中のコメントは適宜読み替えるものとする。

解答群

ア AND GR2,=#7FFF
ウ OR GR2,=#7FFF
オ XOR GR2,=#7FFF

イ AND GR2,=#8000
エ OR GR2,=#8000
カ XOR GR2,=#8000