

# プログラミング演習1

## 課題4: アセンブリ言語

### 第3回 サブルーチン

---

- 座席は自由です
- 講義資料ページを開いておく
- WebClassで出席を登録する準備

# 例題7: サブルーチンの例題

---

## □ 鶴亀算

□ 亀の匹数 = (足の総数 - (2 \* 頭の総数)) / 2

□ 鶴の羽数 = 頭の総数 - 亀の匹数

□ 亀と鶴の合計数は10 ⇒ 頭の総数: 10

□ 足の総数: 26

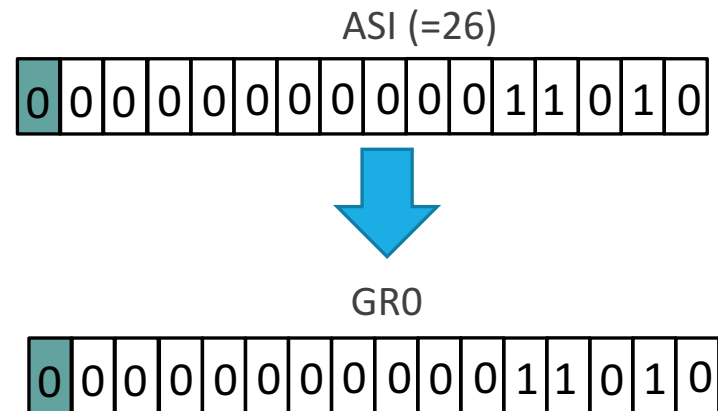


# 例題7: サブルーチンの例題

;例7 鶴亀算 (サブルーチンの使い方)

EX7 START

```
LD GRO,ASI
SUBA GRO,ATAMA
SUBA GRO,ATAMA ;足の総数-頭の総数×2
SRA GRO,1 ;÷2
ST GRO,KAME
LAD GR1,6 ;亀桁位置設定
CALL DECCNV ;10進数に変換
LD GRO,ATAMA
SUBA GRO,KAME
LAD GR1,23 ;鶴桁位置設定
CALL DECCNV ;10進数に変換
OUT OBUF,OLEN
RET
ATAMA DC 10
ASI DC 26
KAME DS 1
OBUF DC 'KAME HIKI:TURU WA'
OLEN DC 31
;
;SUBROUTINE (10進数に変換)
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ
LOOP1 LAD GR2,-1 ;商のカウンタ
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1
SUBA GRO,CONST,GR3
JPL LOOP2
JZE LOOP2
ADDA GRO,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)
ADDA GR2,ZONE ;ゾーン部付加
ST GR2,OBUF,GR1
LAD GR1,1,GR1 ;桁位置のポインタ+1
LAD GR3,1,GR3 ;ループのカウンタ+1
CPA GR3,F5
JMI LOOP1 ;終了?
RET
ZONE DC '0'
CONST DC 10000,1000,100,10,1
F5 DC 5
END
```

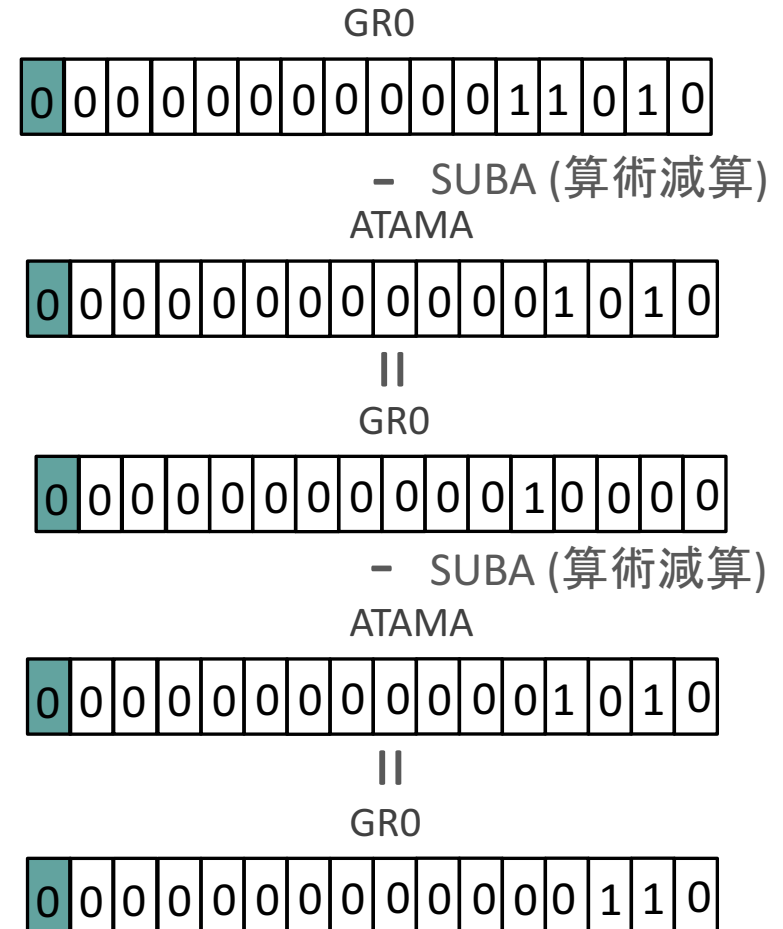


# 例題7: サブルーチンの例題

```

;-----
;例7 鶴亀算 (サブルーチンの使い方)
;-----
EX7 START
LD  GRO,ASI
SUBA GRO,ATAMA
SUBA GRO,ATAMA ;足の総数-頭の総数×2
SRA  GRO,1    ;÷2
ST   GRO,KAME
LAD  GR1,6    ;桁桁位置設定
CALL DECCNV   ;10進数に変換
LD   GRO,ATAMA
SUBA GRO,KAME
LAD  GR1,23   ;鶴桁位置設定
CALL DECCNV   ;10進数に変換
OUT  OBUF,OLEN
RET
ATAMA DC 10
ASI  DC 26
KAME DS 1
OBUF DC 'KAME  HIKI:TURU  WA'
OLEN DC 31
;
;SUBROUTINE (10進数に変換)
DECCNV LAD  GR3,0    ;文字列を数えるループカウンタ
LOOP1  LAD  GR2,-1   ;商のカウンタ
LOOP2  LAD  GR2,1,GR2 ;商のカウンタ+1
SUBA  GRO,CONST,GR3
JPL  LOOP2
JZE  LOOP2
ADDA  GR0,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)
ADDA  GR2,ZONE      ;ゾーン部付加
ST   GR2,OBUF,GR1
LAD  GR1,1,GR1     ;桁位置のポインタ+1
LAD  GR3,1,GR3     ;ループのカウンタ+1
CPA  GR3,F5
JMI  LOOP1         ;終り?
RET
ZONE DC '0'
CONST DC 10000,1000,100,10,1
F5 DC 5
END

```



# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST  GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD  GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

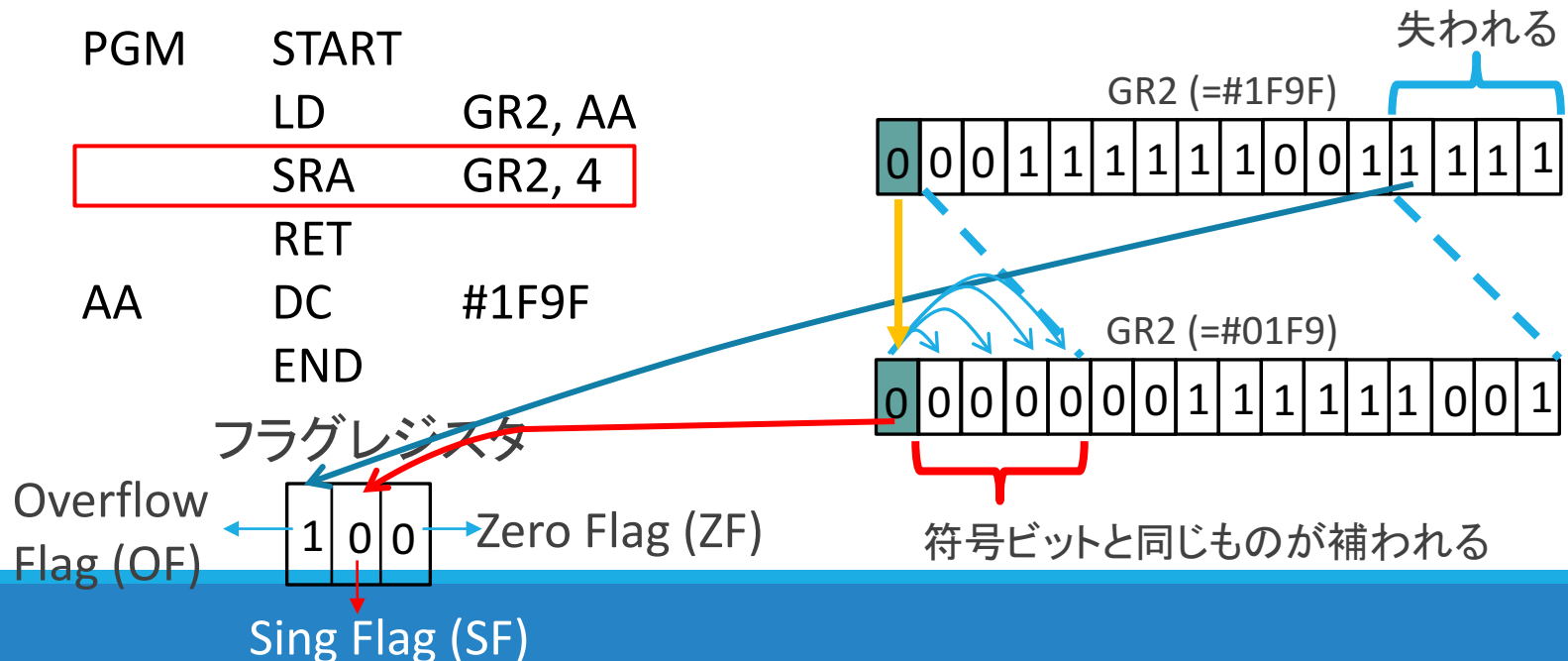
# 例題7: サブルーチンの例題

□ SRA (Shift Right **Arithmetic**; **算術**右シフト) 命令

[ラベル] SRA r, adr[,x]

⇒ rで指定したレジスタの内容を**符号ビットを除き** adr[,x]で指定したビット数だけ右にシフトする

(※シフト後空いたビットには**符号ビットと同じもの**が入る)



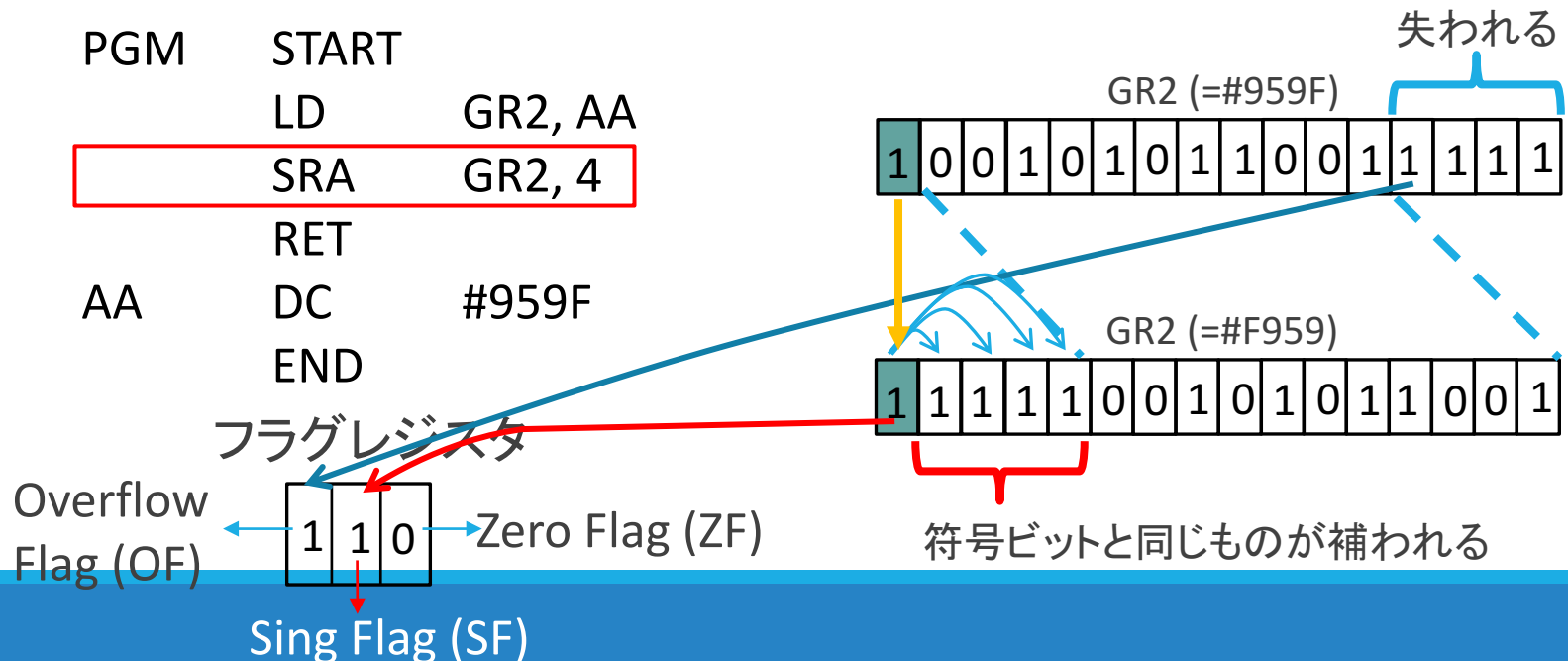
# 例題7: サブルーチンの例題

□ SRA (Shift Right **Arithmetic**; **算術**右シフト) 命令

[ラベル] SRA r, adr[,x]

⇒ rで指定したレジスタの内容を**符号ビットを除き** adr[,x]で指定したビット数だけ右にシフトする

(※シフト後空いたビットには**符号ビットと同じもの**が入る)

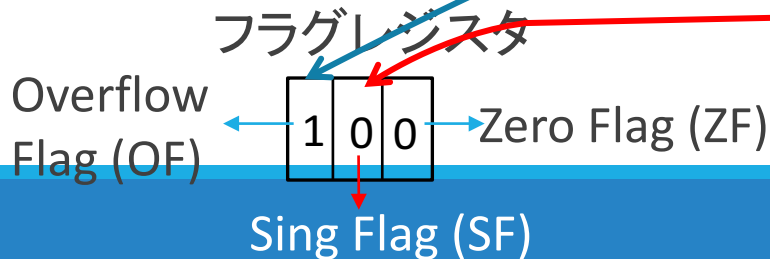
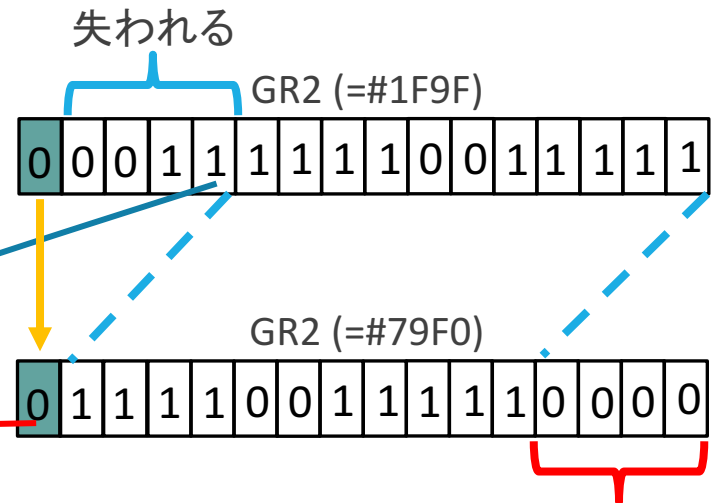


# 例題7: サブルーチンの例題

□ SLA (Shift Left Arithmetic; 算術左シフト) 命令  
 [ラベル] SLA r, adr[,x]

⇒ rで指定したレジスタの内容を符号ビットを除き adr[,x]で指定したビット数だけ左にシフトする  
 (※シフト後空いたビットには0が入る)

```
PGM   START
      LD    GR2, AA
      SLA   GR2, 4
      RET
AA     DC   #1F9F
      END
```



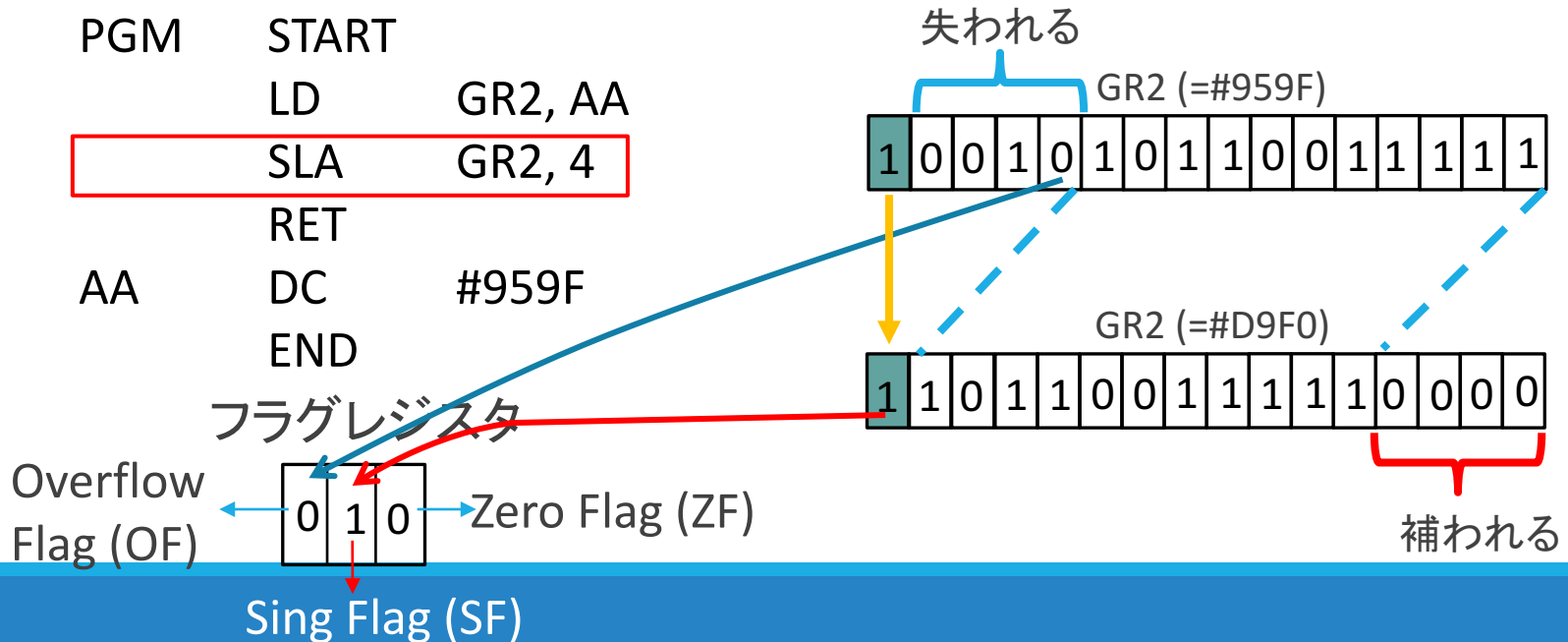
# 例題7: サブルーチンの例題

□ SLA (Shift Left Arithmetic; 算術左シフト) 命令

[ラベル] SLA r, adr[,x]

⇒ rで指定したレジスタの内容を符号ビットを除き adr[,x]で指定したビット数だけ左にシフトする

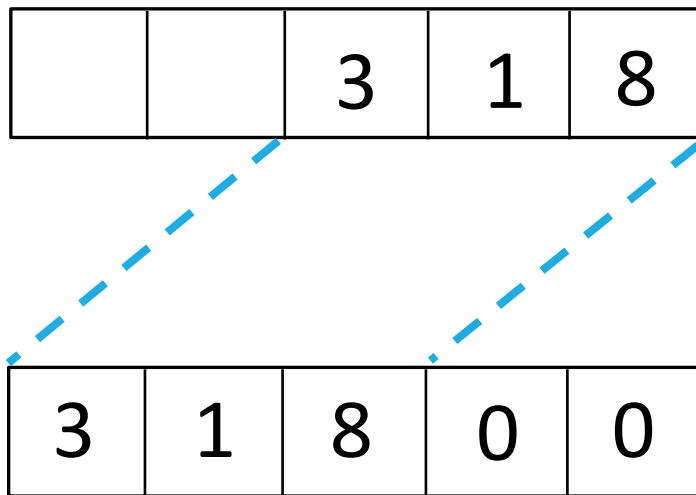
(※シフト後空いたビットには0が入る)



# 例題7: サブルーチンの例題

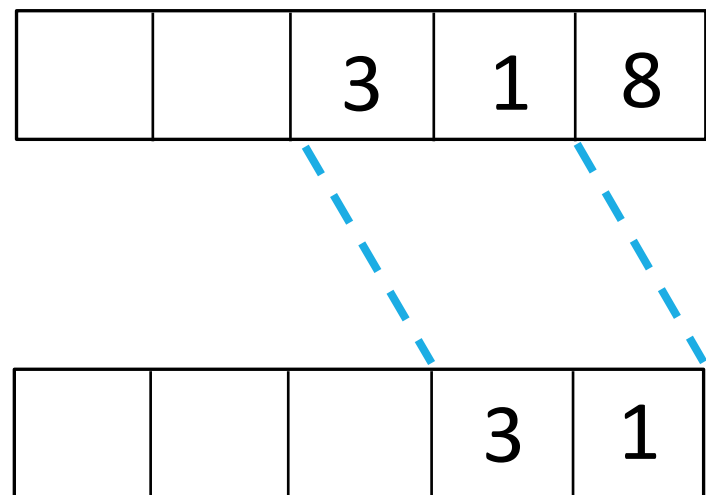
- なぜ除算をビットシフト演算で実現できるのか？
  - 10進数の場合

2桁左に動かすと...



$10^2$ 倍

1桁右に動かすと...

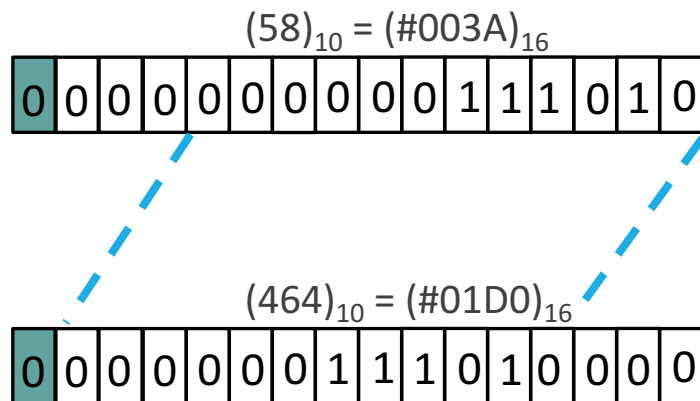


$1/10^1$ 倍  
||  
÷10

# 例題7: サブルーチンの例題

- なぜ除算をビットシフト演算で実現できるのか？
  - 2進数の場合

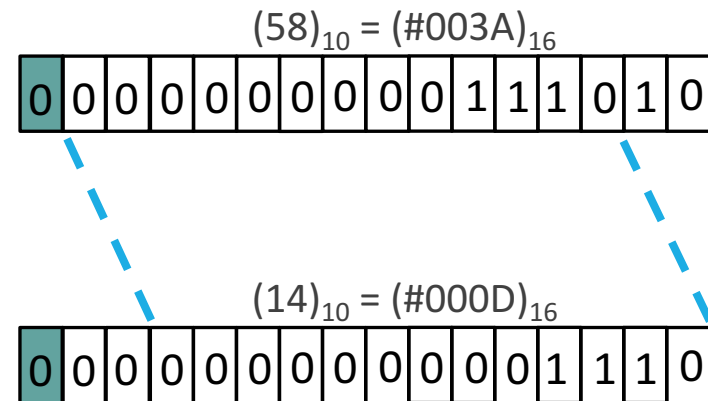
3桁左に動かすと...



$$58 \Rightarrow 464 \text{ (8倍} = 2^3 \text{倍)}$$

n桁左にシフト  $2^n$ 倍

2桁右に動かすと...



$$58 \Rightarrow 14 \text{ (1/4倍} = 1/2^2 \text{倍)}$$

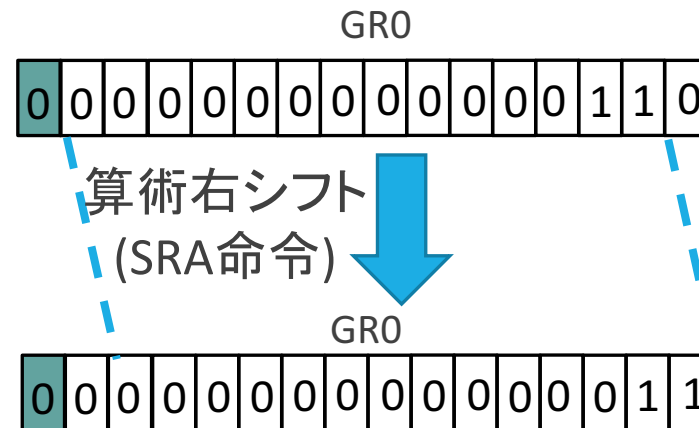
÷4

m桁右にシフト  $1/2^m$ 倍  $\Rightarrow \div 2^m$

# 例題7: サブルーチンの例題

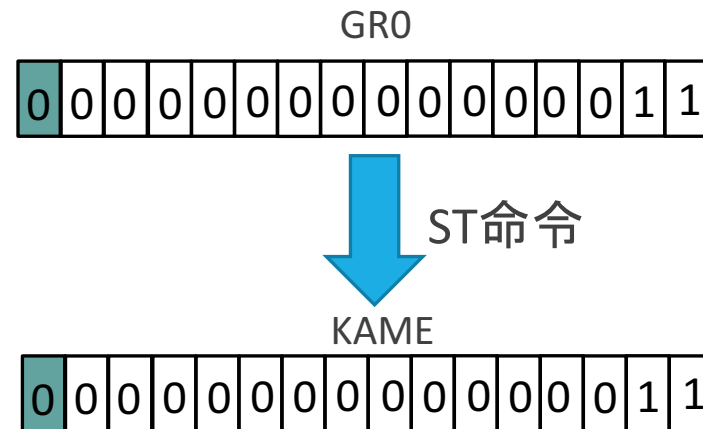
```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA GRO,1 ;÷2  
ST GRO,KAME  
LAD GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD GRO,ATAMA  
SUBA GRO,KAME  
LAD GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

÷2がしたいので、1ビット右シフトすればよい



# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST  GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD  GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```



# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST   GRO,KAME  
LAD  GR1,6 ;亀桁位置設定 → GR1 = 6  
CALL DECCNV ;10進数に変換  
LD   GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI   DC 26  
KAME  DS 1  
OBUF  DC 'KAME  HIKI:TURU  WA'  
OLEN  DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1  LAD GR2,-1 ;商のカウンタ  
LOOP2  LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA  GRO,CONST,GR3  
JPL   LOOP2  
JZE   LOOP2  
ADDA  GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA  GR2,ZONE ;ゾーン部付加  
ST    GR2,OBUF,GR1  
LAD  GR1,1,GR1 ;桁位置のポインタ+1  
LAD  GR3,1,GR3 ;ループのカウンタ+1  
CPA  GR3,F5  
JMI  LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST   GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD   GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

サブルーチンの呼び出し

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST   GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD   GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31
```

```
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GR0,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

➡ 例題6と同じ処理



# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA  GRO,ATAMA  
SUBA  GRO,ATAMA  ;足の総数-頭の総数×2  
SRA  GRO,1      ;÷2  
ST   GRO,KAME  
LAD  GR1,6      ;亀桁位置設定  
CALL  DECCNV    ;10進数に変換  
LD   GRO,ATAMA  
SUBA  GRO,KAME  
LAD  GR1,23     ;鶴桁位置設定  
CALL  DECCNV    ;10進数に変換  
OUT  OBUF,OLEN  
RET  
  
ATAMA DC  10  
ASI  DC  26  
KAME DS  1  
OBUF DC  'KAME  HIKI:TURU  WA'  
OLEN DC  31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD  GR3,0  ;文字列を数えるループカウンタ  
LOOP1  LAD  GR2,-1 ;商のカウンタ  
LOOP2  LAD  GR2,1,GR2 ;商のカウンタ+1  
SUBA  GRO,CONST,GR3  
JPL  LOOP2  
JZE  LOOP2  
ADDA  GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA  GR2,ZONE      ;ゾーン部付加  
ST   GR2,OBUF,GR1  
LAD  GR1,1,GR1     ;桁位置のポインタ+1  
LAD  GR3,1,GR3     ;ループのカウンタ+1  
CPA  GR3,F5  
JMI  LOOP1 ;終了  
RET  
ZONE DC  '0'  
CONST DC  10000,1000,100,10,1  
F5  DC  5  
END
```

  メインルーチンへ戻る

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST   GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD   GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

→ GRO = 10

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA  GRO,ATAMA  
SUBA  GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST  GRO,KAME  
LAD  GR1,6 ;鶴桁位置設定  
CALL  DECCNV ;10進数に変換  
LD  GRO,ATAMA  
SUBA  GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL  DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA  GRO,CONST,GR3  
JPL  LOOP2  
JZE  LOOP2  
ADDA  GRO,CONST,GR3 ;GRO←(GRO)+(CONST+GR3)  
ADDA  GR2,ZONE ;ゾーン部付加  
ST  GR2,OBUF,GR1  
LAD  GR1,1,GR1 ;桁位置のポインタ+1  
LAD  GR3,1,GR3 ;ループのカウンタ+1  
CPA  GR3,F5  
JMI  LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

GRO (=ATAMA,10)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

- SUBA命令  
(算術減算)

KAME (=3)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--



GRO (=7)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
  LD  GRO,ASI  
  SUBA GRO,ATAMA  
  SUBA GRO,ATAMA ;足の総数-頭の総数×2  
  SRA  GRO,1 ;÷2  
  ST  GRO,KAME  
  LAD  GR1,6 ;亀桁位置設定  
  CALL DECCNV ;10進数に変換  
  LD  GRO,ATAMA  
  SUBA GRO,KAME  
  LAD  GR1,23 ;鶴桁位置設定 → GR1 = 23  
  CALL DECCNV ;10進数に変換  
  OUT  OBUF,OLEN  
  RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
  SUBA GRO,CONST,GR3  
  JPL LOOP2  
  JZE LOOP2  
  ADDA GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
  ADDA GR2,ZONE ;ゾーン部付加  
  ST GR2,OBUF,GR1  
  LAD GR1,1,GR1 ;桁位置のポインタ+1  
  LAD GR3,1,GR3 ;ループのカウンタ+1  
  CPA GR3,F5  
  JMI LOOP1 ;終り?  
  RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

# 例題7: サブルーチンの例題

```
-----  
;例7 鶴亀算 (サブルーチンの使い方)  
-----  
EX7 START  
LD  GRO,ASI  
SUBA GRO,ATAMA  
SUBA GRO,ATAMA ;足の総数-頭の総数×2  
SRA  GRO,1 ;÷2  
ST   GRO,KAME  
LAD  GR1,6 ;亀桁位置設定  
CALL DECCNV ;10進数に変換  
LD   GRO,ATAMA  
SUBA GRO,KAME  
LAD  GR1,23 ;鶴桁位置設定  
CALL DECCNV ;10進数に変換  
OUT  OBUF,OLEN  
RET  
ATAMA DC 10  
ASI DC 26  
KAME DS 1  
OBUF DC 'KAME HIKI:TURU WA'  
OLEN DC 31  
;  
;SUBROUTINE (10進数に変換)  
DECCNV LAD GR3,0 ;文字列を数えるループカウンタ  
LOOP1 LAD GR2,-1 ;商のカウンタ  
LOOP2 LAD GR2,1,GR2 ;商のカウンタ+1  
SUBA GRO,CONST,GR3  
JPL LOOP2  
JZE LOOP2  
ADDA GRO,CONST,GR3 ;GR0←(GR0)+(CONST+GR3)  
ADDA GR2,ZONE ;ゾーン部付加  
ST GR2,OBUF,GR1  
LAD GR1,1,GR1 ;桁位置のポインタ+1  
LAD GR3,1,GR3 ;ループのカウンタ+1  
CPA GR3,F5  
JMI LOOP1 ;終り?  
RET  
ZONE DC '0'  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

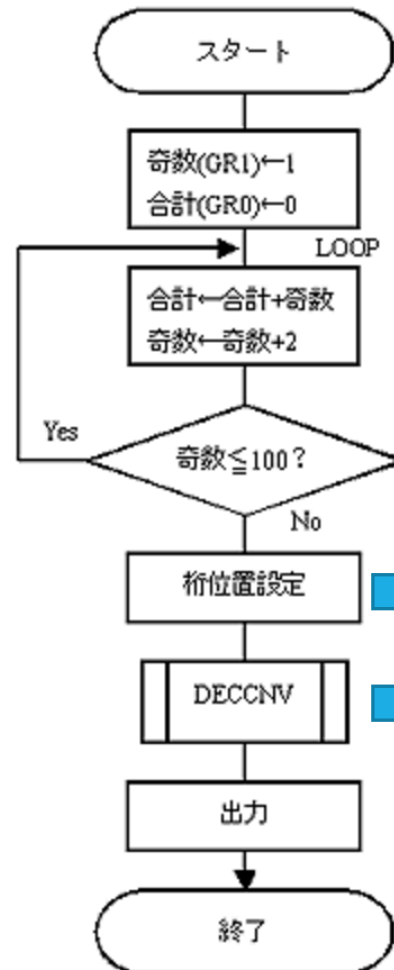


# 例題8:奇数の合計

```

;-----
;例8 奇数の合計
;-----
EX8 START
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1
LOOP  ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )
  LAD GR1,2,GR1 ;GR1 ← 2 + GR1
  CPA GR1,F100  ;GR1とF100を比較 (GR1) - (F100)
  JMI LOOP      ;比較結果が <0 ならLOOPへ分岐
  JZE LOOP      ;比較結果が =0 ならLOOPへ分岐
  LAD GR1,16    ;GR1 ← 16
  CALL DECPRT   ;10進数に変換
  RET
F100 DC 100
;
;SUBROUTINE (10進数に変換)
DECPRT LAD GR3,0 ;GR3 ← 0
LOOP1 LAD GR2,-1 ;GR2 ← -1
LOOP2 LAD GR2,1,GR2 ;GR2 ← 1 + GR2
  SUBA GR0,CONST,GR3 ;GR0 ← ( GR0 ) - ( CONST + GR3 )
  JPL LOOP2 ;減算結果が >0 ならLOOPへ分岐
  JZE LOOP2 ;減算結果が =0 ならLOOP2へ分岐
  ADDA GR0,CONST,GR3 ;GR0 ← ( GR0 ) + ( CONST + GR3 )
  ADDA GR2,ZONE ;GR2 ← ( GR2 ) + ( ZONE )
  ST GR2,OUTBUF,GR1 ;GR2を(OUTBUF + GR1)番地に格納
  LAD GR1,1,GR1 ;GR1 ← 1 + GR1
  LAD GR3,1,GR3 ;GR3 ← 1 + GR3
  CPA GR3,F5 ;GR3とF5を比較 (GR3)-(F5)
  JMI LOOP1 ;比較結果が <0 ならLOOP1へ分岐
  OUT OUTBUF,OUTLEN ;文字を出力する
  RET
ZONE DC '0'
OUTBUF DC 'SUM OF ODD NUM='
OUTLEN DC 21
CONST DC 10000,1000,100,10,1
F5 DC 5
END

```



万, 千, 百, 十, 一の位を  
どこにstoreするか

10進数, 文字列への変換  
(サブルーチン)

# 例題8:奇数の合計

```
-----  
; 例8 奇数の合計  
-----  
EX8 START  
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0  
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1  
  LOOP ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
  LAD GR1,2,GR1  ;GR1 ← 2 + GR1  
  CPA GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
  JMI LOOP       ;比較結果が <0 ならLOOP へ分岐  
  JZE LOOP       ;比較結果が =0 ならLOOP へ分岐  
  LAD GR1,16     ;GR1 ← 16  
  CALL DECPRT    ;10進数に変換  
  RET  
F100 DC 100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD GR3,0 ; GR3 ← 0  
LOOP1 LAD GR2,-1 ; GR2 ← -1  
LOOP2 LAD GR2,1,GR2 ; GR2 ← 1 + GR2  
  SUBA GR0,CONST,GR3 ; GR0 ← ( GR0 ) - ( CONST + GR3 )  
  JPL LOOP2 ; 減算結果が >0 ならLOOP へ分岐  
  JZE LOOP2 ; 減算結果が =0 ならLOOP2 へ分岐  
  ADDA GR0,CONST,GR3 ; GR0 ← ( GR0 ) + ( CONST + GR3 )  
  ADDA GR2,ZONE ; GR2 ← ( GR2 ) + ( ZONE )  
  ST GR2,OUTBUF,GR1 ; GR2を(OUTBUF + GR1)番地に格納  
  LAD GR1,1,GR1 ; GR1 ← 1 + GR1  
  LAD GR3,1,GR3 ; GR3 ← 1 + GR3  
  CPA GR3,F5 ; GR3とF5を比較 (GR3)-(F5)  
  JMI LOOP1 ; 比較結果が <0 ならLOOP1 へ分岐  
  OUT OUTBUF,OUTLEN ; 文字を出力する  
  RET  
ZONE DC '0'  
OUTBUF DC 'SUM OF ODD NUM= '  
OUTLEN DC 21  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

合計 (GR0) = 合計 (GR0) + 奇数 (GR1)

# 例題8:奇数の合計

```
-----  
; 例8 奇数の合計  
-----  
EX8  START  
    LAD  GR0,0      ;合計を格納するレジスタGR0 ← 0  
    LAD  GR1,1      ;奇数を格納するレジスタGR1 ← 1  
LOOP  ADDA  GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
    LAD  GR1,2,GR1 ;GR1 ← 2 + GR1  
    CPA  GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
    JMI  LOOP      ;比較結果が <0 ならLOOP へ分岐  
    JZE  LOOP      ;比較結果が =0 ならLOOP へ分岐  
    LAD  GR1,16     ;GR1 ← 16  
    CALL  DECPRT    ;10進数に変換  
    RET  
F100  DC  100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD  GR3,0    ;GR3 ← 0  
LOOP1  LAD  GR2,-1   ;GR2 ← -1  
LOOP2  LAD  GR2,1,GR2 ;GR2 ← 1 + GR2  
    SUBA  GR0,CONST,GR3 ;GR0 ← ( GR0 ) - ( CONST + GR3 )  
    JPL  LOOP2      ;減算結果が >0 ならLOOP へ分岐  
    JZE  LOOP2      ;減算結果が =0 ならLOOP2 へ分岐  
    ADDA  GR0,CONST,GR3 ;GR0 ← ( GR0 ) + ( CONST + GR3 )  
    ADDA  GR2,ZONE     ;GR2 ← ( GR2 ) + ( ZONE )  
    ST   GR2,OUTBUF,GR1 ;GR2を(OUTBUF + GR1)番地に格納  
    LAD  GR1,1,GR1    ;GR1 ← 1 + GR1  
    LAD  GR3,1,GR3    ;GR3 ← 1 + GR3  
    CPA  GR3,F5       ;GR3とF5を比較 (GR3)-(F5)  
    JMI  LOOP1      ;比較結果が <0 ならLOOP1 へ分岐  
    OUT  OUTBUF,OUTLEN ;文字を出力する  
    RET  
ZONE  DC  '0'  
OUTBUF DC  'SUM OF ODD NUM= '  
OUTLEN DC  21  
CONST  DC  10000,1000,100,10,1  
F5     DC  5  
END
```

奇数 (GR1) に2を足すことで次の奇数を得る

GR1 = 1

GR1 = GR1+2

GR1 = 3

GR1 + 2 = 5

GR1 + 2 = 7 ...

# 例題8:奇数の合計

```
-----  
;例8 奇数の合計  
-----  
EX8 START  
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0  
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1  
LOOP  ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
  LAD GR1,2,GR1  ;GR1 ← 2 + GR1  
  CPA GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
  JMI LOOP       ;比較結果が <0 ならLOOPへ分岐  
  JZE LOOP       ;比較結果が =0 ならLOOPへ分岐  
  LAD GR1,16     ;GR1 ← 16  
  CALL DECPRT    ;10進数に変換  
  RET  
F100 DC 100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD GR3,0 ; GR3 ← 0  
LOOP1 LAD GR2,-1 ; GR2 ← -1  
LOOP2 LAD GR2,1,GR2 ; GR2 ← 1 + GR2  
  SUBA GR0,CONST,GR3 ; GR0 ← ( GR0 ) - ( CONST + GR3 )  
  JPL LOOP2 ; 減算結果が >0 ならLOOPへ分岐  
  JZE LOOP2 ; 減算結果が =0 ならLOOP2へ分岐  
  ADDA GR0,CONST,GR3 ; GR0 ← ( GR0 ) + ( CONST + GR3 )  
  ADDA GR2,ZONE ; GR2 ← ( GR2 ) + ( ZONE )  
  ST GR2,OUTBUF,GR1 ; GR2を(OUTBUF + GR1)番地に格納  
  LAD GR1,1,GR1 ; GR1 ← 1 + GR1  
  LAD GR3,1,GR3 ; GR3 ← 1 + GR3  
  CPA GR3,F5 ; GR3とF5を比較 (GR3)-(F5)  
  JMI LOOP1 ; 比較結果が <0 ならLOOP1へ分岐  
  OUT OUTBUF,OUTLEN ; 文字を出力する  
  RET  
ZONE DC '0'  
OUTBUF DC 'SUM OF ODD NUM= '  
OUTLEN DC 21  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

合計するのは 1 ~ 100 までの奇数

# 例題8:奇数の合計

```
-----  
;例8 奇数の合計  
-----  
EX8 START  
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0  
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1  
  LOOP ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
  LAD GR1,2,GR1  ;GR1 ← 2 + GR1  
  CPA GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
  JMI LOOP       ;比較結果が <0 ならLOOP へ分岐  
  JZE LOOP       ;比較結果が =0 ならLOOP へ分岐  
  LAD GR1,16     ;GR1 ← 16  
  CALL DECPRT   ;10進数に変換  
  RET  
F100 DC 100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD GR3,0 ; GR3 ← 0  
LOOP1 LAD GR2,-1 ; GR2 ← -1  
LOOP2 LAD GR2,1,GR2 ; GR2 ← 1 + GR2  
  SUBA GR0,CONST,GR3 ; GR0 ← ( GR0 ) - ( CONST + GR3 )  
  JPL LOOP2 ; 減算結果が >0 ならLOOP へ分岐  
  JZE LOOP2 ; 減算結果が =0 ならLOOP2 へ分岐  
  ADDA GR0,CONST,GR3 ; GR0 ← ( GR0 ) + ( CONST + GR3 )  
  ADDA GR2,ZONE ; GR2 ← ( GR2 ) + ( ZONE )  
  ST GR2,OUTBUF,GR1 ; GR2を(OUTBUF + GR1)番地に格納  
  LAD GR1,1,GR1 ; GR1 ← 1 + GR1  
  LAD GR3,1,GR3 ; GR3 ← 1 + GR3  
  CPA GR3,F5 ; GR3とF5を比較 (GR3)-(F5)  
  JMI LOOP1 ; 比較結果が <0 ならLOOP1 へ分岐  
  OUT OUTBUF,OUTLEN ; 文字を出力する  
  RET  
ZONE DC '0'  
OUTBUF DC 'SUM OF ODD NUM= '  
OUTLEN DC 21  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

→ GR1 = 101 になるまで加算を繰り返す

# 例題8:奇数の合計

```
-----  
;例8 奇数の合計  
-----  
EX8 START  
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0  
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1  
LOOP  ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
  LAD GR1,2,GR1  ;GR1 ← 2 + GR1  
  CPA GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
  JMI LOOP       ;比較結果が <0 ならLOOP へ分岐  
  JZE LOOP       ;比較結果が =0 ならLOOP へ分岐  
  LAD GR1,16     ;GR1 ← 16  
  CALL DECPRT    ;10進数に変換  
  RET  
F100 DC 100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD GR3,0 ; GR3 ← 0  
LOOP1  LAD GR2,-1 ; GR2 ← -1  
LOOP2  LAD GR2,1,GR2 ; GR2 ← 1 + GR2  
  SUBA GR0,CONST,GR3 ; GR0 ← ( GR0 ) - ( CONST + GR3 )  
  JPL LOOP2          ; 減算結果が >0 ならLOOP へ分岐  
  JZE LOOP2          ; 減算結果が =0 ならLOOP2 へ分岐  
  ADDA GR0,CONST,GR3 ; GR0 ← ( GR0 ) + ( CONST + GR3 )  
  ADDA GR2,ZONE      ; GR2 ← ( GR2 ) + ( ZONE )  
  ST GR2,OUTBUF,GR1 ; GR2を(OUTBUF + GR1)番地に格納  
  LAD GR1,1,GR1      ; GR1 ← 1 + GR1  
  LAD GR3,1,GR3      ; GR3 ← 1 + GR3  
  CPA GR3,F5         ; GR3とF5を比較 (GR3)-(F5)  
  JMI LOOP1         ; 比較結果が <0 ならLOOP1 へ分岐  
  OUT OUTBUF,OUTLEN ; 文字を出力する  
  RET  
ZONE DC '0'  
OUTBUF DC 'SUM OF ODD NUM= '  
OUTLEN DC 21  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

→ 計算結果を格納する桁位置

# 例題8:奇数の合計

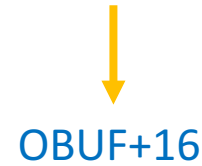
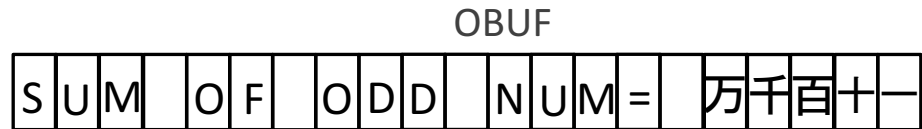
```
-----  
; 例8 奇数の合計  
-----  
EX8  START  
    LAD  GR0,0      ;合計を格納するレジスタGR0 ← 0  
    LAD  GR1,1      ;奇数を格納するレジスタGR1 ← 1  
LOOP  ADDA  GR0,GR1  ;GR0 ← ( GR0 + GR1 )  
    LAD  GR1,2,GR1  ;GR1 ← 2 + GR1  
    CPA  GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
    JMI  LOOP      ;比較結果が <0 ならLOOP へ分岐  
    JZE  LOOP      ;比較結果が =0 ならLOOP へ分岐  
    LAD  GR1,16     ;GR1 ← 16  
    CALL  DECPRT    ;10進数に変換  
    RET  
F100  DC  100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD  GR3,0      ; GR3 ← 0  
LOOP1  LAD  GR2,-1     ; GR2 ← -1  
LOOP2  LAD  GR2,1,GR2  ; GR2 ← 1 + GR2  
    SUBA  GR0,CONST,GR3 ; GR0 ← ( GR0 ) - ( CONST + GR3 )  
    JPL  LOOP2        ; 減算結果が >0 ならLOOP へ分岐  
    JZE  LOOP2        ; 減算結果が =0 ならLOOP2 へ分岐  
    ADDA  GR0,CONST,GR3 ; GR0 ← ( GR0 ) + ( CONST + GR3 )  
    ADDA  GR2,ZONE      ; GR2 ← ( GR2 ) + ( ZONE )  
    ST   GR2,OUTBUF,GR1 ; GR2を(OUTBUF + GR1)番地に格納  
    LAD  GR1,1,GR1     ; GR1 ← 1 + GR1  
    LAD  GR3,1,GR3     ; GR3 ← 1 + GR3  
    CPA  GR3,F5        ; GR3とF5を比較 (GR3)-(F5)  
    JMI  LOOP1        ; 比較結果が <0 ならLOOP1 へ分岐  
    OUT  OUTBUF,OUTLEN ; 文字を出力する  
    RET  
ZONE  DC  '0'  
OUTBUF DC  'SUM OF ODD NUM= '  
OUTLEN DC  21  
CONST  DC  10000,1000,100,10,1  
F5     DC  5  
END
```



# 例題8:奇数の合計

```
-----  
; 例8 奇数の合計  
-----  
EX8 START  
  LAD GR0,0      ;合計を格納するレジスタGR0 ← 0  
  LAD GR1,1      ;奇数を格納するレジスタGR1 ← 1  
LOOP  ADDA GR0,GR1 ;GR0 ← ( GR0 + GR1 )  
  LAD GR1,2,GR1  ;GR1 ← 2 + GR1  
  CPA GR1,F100   ;GR1とF100を比較 (GR1) - (F100)  
  JMI LOOP       ;比較結果が <0 ならLOOP へ分岐  
  JZE LOOP       ;比較結果が =0 ならLOOP へ分岐  
  LAD GR1,16     ;GR1 ← 16  
  CALL DECPRT   ;10進数に変換  
  RET  
F100 DC 100  
;  
;SUBROUTINE (10進数に変換)  
DECPRT LAD GR3,0 ;GR3 ← 0  
LOOP1  LAD GR2,-1 ;GR2 ← -1  
LOOP2  LAD GR2,1,GR2 ;GR2 ← 1 + GR2  
  SUBA GR0,CONST,GR3 ;GR0 ← ( GR0 ) - ( CONST + GR3 )  
  JPL LOOP2          ;減算結果が >0 ならLOOP へ分岐  
  JZE LOOP2          ;減算結果が =0 ならLOOP2 へ分岐  
  ADDA GR0,CONST,GR3 ;GR0 ← ( GR0 ) + ( CONST + GR3 )  
  ADDA GR2,ZONE      ;GR2 ← ( GR2 ) + ( ZONE )  
  ST GR2,OUTBUF,GR1 ;GR2を(OUTBUF + GR1)番地に格納  
  LAD GR1,1,GR1     ;GR1 ← 1 + GR1  
  LAD GR3,1,GR3     ;GR3 ← 1 + GR3  
  CPA GR3,F5        ;GR3とF5を比較 (GR3)-(F5)  
  JMI LOOP1         ;比較結果が <0 ならLOOP1 へ分岐  
  OUT OUTBUF,OUTLEN ;文字を出力する  
  RET  
ZONE DC '0'  
OUTBUF DC 'SUM OF ODD NUM='  
OUTLEN DC 21  
CONST DC 10000,1000,100,10,1  
F5 DC 5  
END
```

GR1 = 16



# 課題

---

## □課題7 (レポート課題)

- 素数かどうかを判断する数に工夫が必要  
⇒ 最小の素数?
- 割り算は減算 (SUBA命令) で実現可能

## □課題8 (レポート課題;任意)

- DATA領域に記憶されている数をどこかにスタック
- データの比較はメインルーチン
- 印刷 (PRINT), 10進数変換 (DECCNV) はサブルーチン