

次の問8は必須問題です。必ず解答してください。

問8 次のプログラムの説明及びプログラムを読んで、設問1～3に答えよ。

整数型関数 BitTest は、8ビットのデータ中の指定したビット位置にあるビットの値を検査して、結果を返す。整数型関数 BitCount は、8ビットのデータ中にある1のビットの個数を返す。

なお、本問において、演算子“&”，“|”は、二つの8ビット論理型データの対応するビット位置のビット同士について、それぞれ論理積、論理和を求め、8ビット論理型で結果を得るものとする。また、"～"B という表記は、8ビット論理型定数を表す。

[プログラム1の説明]

整数型関数 BitTest を、次のとおりに宣言する。

○整数型関数 : BitTest (8ビット論理型 : Data, 8ビット論理型 : Mask)

検査される8ビットのデータは入力用の引数 Data に、検査をするビット位置の情報は入力用の引数 Mask に、それぞれ格納されている。Mask 中のビットの値が1であるビット位置に対応した Data 中のビットを検査して、次の返却値を返す。ここで、Mask 中には1のビットが1個以上あるものとする。

返却値 0 : 検査した全てのビットが0

1 : 検査したビット中に0と1が混在

2 : 検査した全てのビットが1

例えば、図1の例1では、Mask のビット番号7～5の3ビットが1であるので、Data のビット番号7～5の3ビットの値を検査し、0と1が混在しているので返却値1を返す。例2では、Mask のビット番号4と0の2ビットが1であるので、Data のビット番号4と0の2ビットの値を検査し、どちらも1であるので返却値2を返す。

(例1)	(例2)
ビット番号 7 6 5 4 3 2 1 0	ビット番号 7 6 5 4 3 2 1 0
Data 0 1 0 1 0 1 0 1	Data 0 0 1 1 0 0 1 1
Mask 1 1 1 0 0 0 0 0	Mask 0 0 0 1 0 0 0 1
返却値 1	返却値 2

図1 BitTest の実行例

〔プログラム1〕

- 整数型関数 : BitTest (8 ビット論理型 : Data, 8 ビット論理型 : Mask)
- 整数型 : RC /* 返却値 */
 - RC ← 2 /* 返却値は 2 */
 - RC ← 0 /* 返却値は 0 */
 - RC ← 1 /* 返却値は 1 */
 - return RC /* RC を返却値として返す */

[プログラム2, 3の説明]

整数型関数 BitCount を、次のとおりに宣言する。

○整数型関数: BitCount (8 ビット論理型: Data)

検査される 8 ビットのデータは入力用の引数 Data に格納されている。

このためのプログラムとして、基本的なアルゴリズムを用いたプログラム 2 と、処理効率を重視したプログラム 3 を作成した。

プログラム 2, 3 中の各行には、ある処理系を想定して、プログラムの各行を 1 回実行するときの処理量 (1, 2, ...) を示してある。選択処理と繰返し処理の終端行の処理量は、それぞれの開始行の処理量に含まれるものとする。

なお、演算子“-”は、両オペランドを8ビット符号なし整数とみなして、減算を行うものとする。

[プログラム 2]

(处理量)

- 整数型関数 : BitCount (8 ビット論理型 : Data)
- 8 ビット論理型 : Work
- 整数型 : Count, Loop

1 • Work \leftarrow Data

1 • Count \leftarrow 0

4 ■ Loop: 0, Loop $<$ 8, 1

3 ▲ Work の最下位ビットが 1

1 • Count \leftarrow Count + 1

1 ▼

1 • Work を右へ 1 ビット論理シフトする

2 • return Count /* Count を返却値として返す */

[プログラム 3]

(処理量)

○整数型関数 : BitCount (8 ビット論理型 : Data)
○8 ビット論理型 : Work
○整数型 : Count

```
1   • Work ← Data
1   • Count ← 0
2   ■ Work 中に 1 のビットがある
1   • Count ← Count + 1
3   • Work ← Work & (Work - 1) ←————  $\alpha$ 
2   • return Count      /* Count を返却値として返す */
```

設問 1 プログラム 1 中の に入れる正しい答えを、解答群の中から選べ。

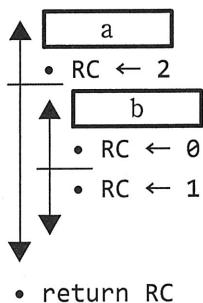
解答群

ア (Data & Mask) = "00000000"B イ (Data & Mask) = Data
ウ (Data & Mask) = Mask エ (Data | Mask) = "00000000"B
オ (Data | Mask) = Mask

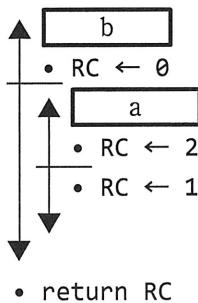
設問 2 次の記述中の に入れる正しい答えを、解答群の中から選べ。

プログラム 1 は、Mask 中に 1 のビットが 1 個以上あることを前提としている。ここで、この前提を取り除いて、Mask 中の 1 のビットが 0 個の場合は返却値 0 を返すようにしたい。そのため、プログラム 1 の処理部分について、次の修正案①～③を考えた。ここで、修正案①は、プログラム 1 のままで何も変更しない。また、 a と b には、設問 1 の正しい答えが入っているものとする。

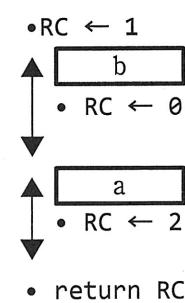
修正案①(変更なし)



修正案②



修正案③



これらの修正案のうち、正しく動作するのは である。

解答群

ア 修正案①

イ 修正案②

ウ 修正案③

エ 修正案①及び②

オ 修正案①及び③

カ 修正案②及び③

設問3 次の記述中の に入れる正しい答えを、解答群の中から選べ。

プログラム2, 3の処理効率について考えてみる。表1にプログラム2, 3の処理量の比較結果を示す。

表1 プログラム2, 3の処理量の比較

	最小	最大
プログラム2	72	<input type="text"/> d
プログラム3	<input type="text"/> e	54

プログラム3では、 α の行での変数Workの更新において効率の良いアルゴリズムが使われている。例えば、プログラム3で引数Dataの内容が"01101010"Bであったとき、繰返し処理において α の行の2回目の実行が終了した時点で変数Workの内容は、" f "Bになっている。このようなビット変換の処理によって、繰返し処理の繰返し回数は、検査されるデータ中の1のビットの個数と同じになる。

d に関する解答群

ア 80

イ 88

ウ 104

エ 112

e に関する解答群

ア 6

イ 10

ウ 20

エ 22

f に関する解答群

ア 00000011

イ 00000110

ウ 00001010

エ 01010000

イ 01100000

カ 10100000