

問題 4 次のプログラムの説明および擬似言語の記述形式の説明を読み、プログラム中の  
[ ]に入れるべき適切な字句を解答群から選べ。

[プログラムの説明]

図 1 のレコード形式の売上ファイル(u-file)を入力し、分野コード別、商品コード別に売上数量の合計を求めて出力するものである。

なお、売上ファイルには、同一の分野コードと商品コードを持つレコードが複数存在する場合がある。

また、分野別コード、商品別コードの昇順に整列されている。

分野コード	商品コード	売上数量
bcode	scode	suryo

図 1 売上ファイルのレコード

出力は、図 2 のように、1 レコードごとの情報を出力しながら、商品コードが変わった場合は商品別合計を出力し、分野コードが変わった場合は商品別合計と分野別合計を出力する。

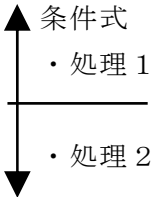

また、最後に売上数量の総合計を出力する。

分野コード	商品コード	売上数量
A001	S100200	120
A001	S100200	100
	商品別合計	220
A001	S101100	80
A001	S101100	20
	商品別合計	100
	分野別合計	320
A101	S002301	12
⋮	⋮	⋮
X210	S290032	120
	商品別合計	350
	分野別合計	860
	総合計	13,256

図 2 出力例

なお、プログラム中で用いている最高値とは、コンピュータで表現できる一番大きな値であり、分野コードおよび商品コードに含まれていない。

[擬似言語の記述形式の説明]

記述形式	説明
○	手続き，変数などの名前，型などを宣言する
・変数 ← 式	変数に式の値を代入する
/* 文 */	注釈を記述する
	<p>選択処理を示す。</p> <p>条件式が真の時は処理 1 を実行し， 偽の時は処理 2 を実行する。</p>
	<p>前判定繰り返し処理を示す。</p> <p>条件式が真の間，処理を実行する。</p>

<次頁へ続く>

[プログラム]

○プログラム名: **Shukei()**

○整数型: **suryo, a\_ttl, b\_ttl, s\_ttl**

○文字型: **bcode, scode, old\_bcode, old\_scode**

○ファイル: **u-file**

○手続き: **FileRead(u-file, bcode, scode, suryo)**

/\* 売上ファイル (u-file) から1レコードずつ読み込み \*/  
/\* 分野コードを bcode, 商品コードを scode, 売上数量を \*/  
/\* suryo に設定する。ファイルの最後を示す識別子を入力 \*/  
/\* した場合は, bcode と scode に最高値が設定される。 \*/

・ ファイルを開く

・ **FileRead(u-file, bcode, scode, suryo)**

・ **a\_ttl ← 0**

■ **bcode ≠ 最高値**

・ **b\_ttl ← 0**

・ (1)

■ **bcode = old\_bcode**

・ **s\_ttl ← 0**

・ (2)

■ **bcode = old\_bcode かつ scode = old\_scode**

・ **bcode, scode, suryo** を出力

・ (3)

・ **FileRead(u-file, bcode, scode, suryo);**

■

・ "商品別合計", **s\_ttl** を出力

・ (4)

■

・ "分野別合計", **b\_ttl** を出力

・ (5)

■

・ "総合計", **a\_ttl** を出力

・ ファイルを閉じる

(1) , (2) の解答群

ア. `bcode ← scode`

ウ. `old_bcode ← bcode`

オ. `old_scode ← bcode`

イ. `scode ← bcode`

エ. `old_bcode ← scode`

カ. `old_scode ← scode`

(3) ~ (5) の解答群

ア. `a_ttl ← a_ttl + b_ttl`

ウ. `a_ttl ← a_ttl + suryo`

オ. `b_ttl ← b_ttl + s_ttl`

キ. `s_ttl ← s_ttl + a_ttl`

ケ. `s_ttl ← s_ttl + suryo`

イ. `a_ttl ← a_ttl + s_ttl`

エ. `b_ttl ← b_ttl + a_ttl`

カ. `b_ttl ← b_ttl + suryo`

ク. `s_ttl ← s_ttl + b_ttl`