

1

配列操作<数値型の配列>

問 1-1

次のCプログラムの説明及びプログラムを読んで、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。

[プログラムの説明]

主プログラムから渡された正の整数 num (4,095 以下) を4けたの8進数に変換した結果を、配列 octal (要素数: 4個) に格納するプログラムである。

num \longrightarrow octal

[0]	[1]	[2]	[3]
5	3	0	7

[プログラム]

```
void ChangeOctal(int num, int octal[])
{
    int base[] = {512, 64, 8, 1}, i;

    for(  ) {
        octal[i] =  ;
        num -= octal[i] * base[i];
    }
}
```

aに関する解答群

- | | |
|------------------------|------------------------|
| ア i = 0 ; i < 4 ; i++ | イ i = 1 ; i < 5 ; i++ |
| ウ i = 3 ; i >= 0 ; i-- | エ i = 4 ; i >= 1 ; i-- |

bに関する解答群

- | | |
|----------------------|----------------------|
| ア base[i] / num | イ base[i] / octal[i] |
| ウ num / base[i] | エ num / octal[i] |
| オ octal[i] / base[i] | カ octal[i] / num |

問 1-2

次のCプログラムの説明及びプログラムを読んで、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。

[プログラムの説明]

50 種類の商品の在庫数を更新するプログラムである。

- (1) 商品の昨日時点の在個数が配列 Szaiko, 今日の入庫数が配列 Izaiko, 今日の出庫数が配列 Ozaiko に格納されている。

なお、各配列の要素は同一要素番号で対応し、入出庫がなかった場合には0が格納されている。

Szaiko	Izaiko	Ozaiko
[0] <input type="text"/>	[0] <input type="text"/>	[0] <input type="text"/>
[1] <input type="text"/>	[1] <input type="text"/>	[1] <input type="text"/>
[2] <input type="text"/>	[2] <input type="text"/>	[2] <input type="text"/>
⋮ <input type="text"/>	⋮ <input type="text"/>	⋮ <input type="text"/>

- (2) プログラムは、昨日時点の在庫数を今日の入庫数と出庫数で更新して、今日時点の在庫数を配列 Szaiko に求める。ただし、更新後の在庫数の値が負になった場合はトランザクションエラーと考え、昨日時点の在庫数を今日時点の在庫数とする。

[プログラム]

```
void ZaikoUpdate(int Szaiko[], int Izaiko[], int Ozaiko[])
{
    int i;

    for(i = 0 ; i < 50 ; i++) {
        Szaiko[i] =  a ;
        if(Szaiko[i] < 0)
            Szaiko[i] =  b ;
    }
}
```

a, bに関する解答群

- | | |
|-------------------------------------|-------------------------|
| ア 0 | イ Izaiko[i] + Ozaiko[i] |
| ウ Izaiko[i] - Ozaiko[i] | エ Ozaiko[i] - Izaiko[i] |
| オ Szaiko[i] | カ Szaiko[i] * (-1) |
| キ Szaiko[i] + Izaiko[i] + Ozaiko[i] | |
| ク Szaiko[i] + Izaiko[i] - Ozaiko[i] | |
| ケ Szaiko[i] - Izaiko[i] + Ozaiko[i] | |
| コ Szaiko[i] - Izaiko[i] - Ozaiko[i] | |

問 1-3 【平成 10 年秋 二種問 7】

次の C プログラムの説明及びプログラムを読んで、設問に答えよ。

[プログラムの説明]

自動販売機に投入された金額と商品の価格の差額から、釣銭として返却する各貨幣の枚数を計算するプログラムである。

- (1) 商品の価格は、引数 price に格納されている。
なお、商品の価格には、10円未満の端数はない。
- (2) 投入された各貨幣の枚数は、配列 input に格納されている。
なお、投入された金額が、商品の価格を下回ることはない。

input[0]	1000円貨幣の枚数
[1]	500円貨幣の枚数
[2]	100円貨幣の枚数
[3]	50円貨幣の枚数
[4]	10円貨幣の枚数

- (3) 釣銭として返却する各貨幣の枚数を、配列 output に格納する。このとき、返却する貨幣の合計枚数をできるだけ少なくする。

output[0]	1000円貨幣の枚数
[1]	500円貨幣の枚数
[2]	100円貨幣の枚数
[3]	50円貨幣の枚数
[4]	10円貨幣の枚数

- (4) 自動販売機が保持している釣銭用の貨幣の枚数は、貨幣の種類別に、大域変数である配列 stock に格納される。stock[0] の初期値は 0、それ以外の stock の要素の初期値は 10 とする。

stock[0]	1000円貨幣の枚数
[1]	500円貨幣の枚数
[2]	100円貨幣の枚数
[3]	50円貨幣の枚数
[4]	10円貨幣の枚数

- (5) 投入された貨幣は、次回以降の釣銭として用いるために、配列 stock に加える。
- (6) 戻り値として次の値を返す。
 - −1：釣銭切れの場合
 - 0：釣銭不要の場合
 - 1：釣銭を返却する場合
- (7) 釣銭切れの場合、投入された各貨幣の枚数を配列 output に格納する。

[プログラム]

```
#define TBLSIZE 5

static int  money[] = {1000, 500, 100, 50, 10},
           stock[] = {0, 10, 10, 10, 10};

int VendingMachine(int price, int input[], int output[])
{
    int change, k, paid, retval;

    paid = 0;
    for(k = 0 ; k < TBLSIZE ; k++) {
        paid += input[k] * money[k];
    }
    change = paid - price
    if (change != 0) {
        for(  ) {
            output[k] =  ;
            if (output[k] > stock[k])
                output[k] = stock[k];
            change -= output[k] * money[k];
        }
        if (  ) {
            for(k = 0 ; k < TBLSIZE ; k++)
                output[k] = input[k];
            retval = -1;
        } else {
            for(k = 0 ; k < TBLSIZE ; k++)
                stock[k] =  ;
            retval = 1;
        }
    } else {
        for(k = 0 ; k < TBLSIZE ; k++) {
            output[k] = 0;
            stock[k] += input[k];
        }
        retval = 0;
    }
    return retval;
}
```