2013年度センター試験　情報関係基礎

第3問

　Ｂさんは、24時間営業の飲食店を多数経営している。店ごとに、一日の時間帯別の利益を集計したところ、損失となる時間帯があることがわかったので、24時間営業をやめ営業時間を見直すことにした。
　表1に、ある店の午前2時から翌日午前2時までの間を3時間ごとに区切った時間帯別の、一日あたりの利益を示す。利益の単位は千円とし、負の利益は、人件費などの支出が売上金額を上回り、損失が出たことを意味する。

表1　ある店の時間帯別の一日あたりの利益

時間帯	1	2	3	4	5	6	7	8
時　間	2 - 5	5 - 8	8 - 11	11 - 14	14 - 17	17 - 20	20 - 23	23 – 2
利　益	1	−3	2	10	−4	12	8	−4

　店ごとに、次の条件を満たす営業時間を求める。このような営業時間は一通りに定まるものと仮定する。
・営業時間は、午前2時から翌日午前2時までの範囲内の連続時間とし、営業時間中は一時閉店しない。
・営業時間内の利益の和（以下、総利益という）が最大である。この最大値は正であると仮定する。
　以下、時間帯 i の開始時刻から時間帯 j の終了時刻までの総利益を $[i,\ j]$ と書くことにする。ただし $i \leq j$ とする。

表２　表１の店に関する $[i,\ j]$ の一覧表

		終了時間帯 j							
		1	2	3	4	5	6	7	8
開始時間帯 i	1	1	-2	0	10	6	18	26	22
	2		**アイ**	－1	9	5	17	25	21
	3			2	12	8	20	28	24
	4				10	6	18	**ウエ**	22
	5					－4	**オ**	16	12
	6						12	20	16
	7							8	4
	8								－4

問１　Ｒさんは、ためしに表１の店について $[i,\ j]$ を手計算で求め、表２（前ページ）を作成してみた。たとえば、$[1,\ 3]$ は時間帯１から時間帯３までの利益の和である $1+(-3)+2=0$ が記入される。

$i=j$ であるときはその時間帯の利益のみ記入されるため、**アイ** には $\mathbf{-3}$ が当てはまる。以下 **ウエ** 、**オ** を求めると

ウエ ：$[4,\ 7]=10+(-4)+12+8=\mathbf{26}$
オ ：$[5,\ 6]=\ -4+12=\mathbf{8}$

となる。すると、$[3,\ 7]=28$ が総利益の最大値なので、この店の営業時間を時間帯３から時間帯７まで、すなわち８時から23時までに変更すればよいことがわかった。

問１の正解

ア	イ	ウ	エ	オ
－	3	2	6	8

問２　問 1 では総利益が最大となる営業時間を手計算で求めたが、経営する店が多数あるので、この方法を自動化する必要があった。そこで図１に示す手続きを作成し、それぞれの店の営業時間を決定することにした。表 1 のような時間帯別の利益は、時間帯番号を添え字とする配列 **Rieki** にあらかじめ格納されている。また、求まった営業時間の最初と最後の時間帯をそれぞれ変数 **kaisi** と **syuryo** に、総利益の最大値を変数 **saidaiRieki** に格納する。

```
saidaiRieki ← 0
i を [ク] から 8 まで 1 ずつ増やしながら、
|  soRieki ← 0
|  j を [ケ] から 8 まで 1 ずつ増やしながら、
|  |  soRieki ← [コ]
|  |  もし [サ] ならば
|  |  |  saidaiRieki ← soRieki
|  |  |  kaisi ← [シ]  syuryo ← [ス]
|  |  を実行する
|  を繰り返す
を繰り返す
「開始時間帯は」と kaisi と「とし、」を表示する
「終了時間帯は」と syuryo と「とし、」を表示する
「総利益の最大値は」と saidaiRieki と「千円である。」を表示する
```

図１　総利益が最大となる営業時間を求める手続き１

図１の手続きは以下の手順を行っている。

　まず、総利益 [1, 1] を求める。今の時点でこの総利益が最大である。

→ 次に [1, 1] の値に時間帯２の利益を加え、総利益[1, 2] を求める。

→ [1, 1] と [1, 2] を比較し、大きい値を最大利益とする。また、最大となる開始時間帯と終了時間帯も記録する。

→ 同様にして各時間帯を順に加え [1, 3], [1, 4], …,[1, 8] を求め、これまでの最大利益より大きくなるか判定する。新たな最大利益を得た場合は、開始時間帯と終了時間帯も変更する。

→ 続いて総利益 [2, 2], [2, 3], … , [2, 8] を順に求めて、最大利益であるかの判定を行う。これを [8, 8] まで行う。

この手順に基づいて、図１の空欄を考えていく。

(02) 行の変数 i は開始時間帯を表すため、「i を ① 1 から 8 まで 1 ずつ増やし」ていく。
(03) 行の変数 soRieki は開始時間帯が i であるときの総利益を表す。
(04) 行の変数 j は終了時間帯を表すため、時間帯は i 以降でなければならない。よって「j を ② i から 8 まで 1 ずつ増やし」ていく。
(05) 行で総利益 $[i,\ j]$ を求めていく。そのためには Rieki[j] を順に足していけばよい。よって (05) 行の内容は

soRieki ← （b）soRieki + soRieki[j]

となる。これにより soRieki には $[i,\ j]$ の値が入力されている。
(06) 行で (05) 行で求められた soRieki=$[i,\ j]$ が最大であるかの判定を行う。そのため「もし ② soRieki> saidaiRieki ならば」となる。
(07) 行は前の行で soRieki> saidaiRieki であるとき、最大利益を変更する。
(08) 行は開始時間帯と終了時間帯を変更する。よって「kaisi ← ② i　syuryo ← ③ j」となる。

(10) 行で、$[i,\ i]$ から $[i,\ 8]$ までの最大利益を求めていく。
(11) 行で、i を 1 増やして最大利益を求めていく。

この手続き１の行 (06) にある soRieki>saidaiRieki の比較はすべての $[i,\ j]$ に対して比較を行うため、全部で $8+7+\cdots+1=\mathbf{36}$ 回行われる。

図１の手続き１の空欄を埋めたものは以下の通りになる。

```
saidaiRieki ← 0
i を 1 から 8 まで 1 ずつ増やしながら、
| soRieki ← 0
| j を i から 8 まで 1 ずつ増やしながら、
| | soRieki ← soRieki+Rieki[j]
| | もし soRieki > saidaiRieki ならば
| | | saidaiRieki ← soRieki
| | | kaisi ← i  syuryo ← j
| | を実行する
| を繰り返す
を繰り返す
「開始時間帯は」と kaisi と「とし、」を表示する
「終了時間帯は」と syuryo と「とし、」を表示する
「総利益の最大値は」と saidaiRieki と「千円である。」を表示する
```

問２の正解

カ	3	キ	6	ク	1	ケ	2	コ	b	サ	2	シ	2	ス	3

問 3

Rさんは表 2 において、$[1,\ 3]$ ～ $[1,\ 8]$ がそれぞれ $[3,\ 3]$ ～ $[3,\ 8]$ より 2 だけ小さいのは、$[1,\ 2] = -2$ が理由であることに気がついた。同じように、$[6,\ 6]$ ～ $[6,\ 8]$ がそれぞれ $[3,\ 6]$ ～ $[3,\ 8]$ より 8 だけ小さいのは、$[3,\ 5] = 8$ が理由である。

一般的には、$i \leq j < k$ とするとき、$[i,\ k] = [i,\ j] + [j+1,\ k]$ から、次がわかる。

(1) $[i,\ j] \geq 0$ のとき、$[i,\ k] \geq [j+1,\ k]$ なので、$[j+1,\ k]$ を計算する必要はない。

(2) $[i,\ j] < 0$ のとき、$[i,\ k] < [j+1,\ k]$ なので、$[i,\ k]$ を計算する必要はない。

これらの性質を利用して、$[1,\ 2],\ [1,\ 4],\ [3,\ 6],\ [6,\ 7]$ の計算の必要性を調べてみる。

$[1,\ 2]$: $[1,\ 1] = 1 \geq 0$ であるため、$[1,\ 2]$ がより大きいことは示されていない。

$[1,\ 4]$: $[1,\ 2] = -2 < 0$ であるため、(2) より $[1,\ 4]$ を計算する必要はない。

$[3,\ 6]$: $[i,\ 2] < 0$ $(i = 1,\ 2)$ また $[3,\ j] \geq 0$ $(j = 3,\ 4,\ 5)$ であるため、$[3,\ 6]$ がより大きいことは示されていない。

$[6,\ 7]$: $[1,\ 5] = 6 \geq 0$ より、(1) から $[6,\ 7]$ を計算する必要はない。

よってこれらの性質を利用すれば、表 2 の ④ $[1,\ 4],\ [6,\ 7]$ などは計算しなくても、総利益の最大値を求めることができる。この考えに基づいて、図 1 の手続きを改良しよう。

上の (1) から、計算中の総利益が 0 以上である間は、終了時間帯を次にずらして総利益を求める。(2) から、計算中の総利益が負になった場合には、そのときの終了時間帯の次を開始時間帯に設定して先に調べていく。

<table>
<tr><th>開始
時間帯</th><th>終了
時間帯</th><th>総利益</th><th>総利益の
最大値</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>2</td><td>－2</td><td>1</td></tr>
<tr><td colspan="4">総利益が負になったため、
開始時間帯を 3 に設定する。</td></tr>
<tr><td>3</td><td>3</td><td>2</td><td>2</td></tr>
<tr><td>3</td><td>4</td><td>12</td><td>12</td></tr>
<tr><td>3</td><td>5</td><td>8</td><td>12</td></tr>
<tr><td>3</td><td>6</td><td>20</td><td>20</td></tr>
<tr><td>3</td><td>7</td><td>28</td><td>28</td></tr>
<tr><td>3</td><td>8</td><td>24</td><td>28</td></tr>
<tr><td colspan="4">終了時間帯が 8 になり、0 以上で
あったため、以降の総利益の計算
の必要はない。</td></tr>
</table>

この考えに基づき、表 1 に対する処理過程を示すと、下の表 3 の（a）→（b）→ … →（h）のようになる。ここで、いくつかの値を「？」で隠してある。これらの値も以下の通りになる。

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
開始時間帯	1	? = 1	**3**	? = 3	? = 3	? = 3	3	3
終了時間帯	1	2	3	4	5	6	7	8
総利益	1	?=－2	2	? = 12	8	**20**	28	24
総利益の最大値	1	1	2	**12**	? = 12	? = 20	28	28

この考えに従って、図１を改良した手続きを図２に示す。

```
(01)  saidaiRieki ← 0, soRieki ← 0, i ← 1
(02)  j を 1 から 8 まで 1 ずつ増やしながら、
(03)  |  soRieki ← ナ
(04)  |  もし サ ならば
(05)  |  |  saidaiRieki ← soRieki
(06)  |  |  kaisi ← ニ , syuryo ← ス
(07)  |  を実行し、そうでなくてもし soRieki ＜ 0 ならば
(08)  |  |  soRieki ← ヌ , i ← ネ
(09)  |  を実行する
(10)  を繰り返す
(11)  「開始時間帯は」と kaisi と「とし、」を表示する
(12)  「終了時間帯は」と syuryo と「とし、」を表示する
(13)  「総利益の最大値は」と saidaiRieki と「千円である。」を表示する
```

図２　総利益が最大となる営業時間を求める手続き２

図２の手続きは以下の手順を行っている。

まず、総利益 $[1,\ 1]$ を求める。

→ $[1,\ 1]$ が正であれば、最大総利益を変更する。

もし、$[1,\ 1]$ が負であるならば、$[1,\ 2]$ 以降の計算を行わず、開始時間帯を２に変えて、総利益の計算を続ける。

一方、$[1,\ 1]$ が０以上であるならば、続けて $[1,\ 2]$ の総利益の計算を続ける。

→ 次の $[i,\ j]$ について、最大総利益であるかの判定を行い、新たな最大総利益が求まるときは、開始時間帯と終了時間帯を変更する。

$[i,\ j]$ が負であるときは、開始時間帯を $j+1$ に変えて、総利益の計算を続ける。

一方、$[i,\ j]$ が０以上であるならば、続けて $[i,\ j+1]$ の総利益の計算を続ける。これを $j=8$ になるまで行う。

この手順にもとづいて図２の手続きを考える。

(03) 行は総利益 $[i,\ j]$ を求めるため「soRieki ← (d) soRieki+Rieki[j]」となる。

(04) 行で総利益 $[i,\ j]$ が最大であるかの判定を行う。新たな最大総利益が求ま

るときは、(05) 行で最大総利益の変更をして、(06) 行では開始時間帯と終了時間帯の変更をする。そのため、「kaisi ← ③ i , syuryo ← j」となる。
(07) 行で $[i,\ j]$ が負であるかの判定を行う。負であるときは、(08) 行で開始時間帯を $j+1$ に変えて総利益の計算を行う。また総利益を改めて 0 にする必要がある。このことから (08) 行は「soRieki ← ⓪ 0, i← ⑨ j+1」となる。

図２の手続き２の空欄を埋めたものは以下の通りになる。

```
(01)  saidaiRieki ← 0, soRieki ← 0, i ← 1
(02)  j を 1 から 8 まで 1 ずつ増やしながら、
(03)  |  soRieki ← soRieki+ Rieki[j]
(04)  |  もし soRieki > saidaiRieki ならば
(05)  |  |  saidaiRieki ← soRieki
(06)  |  |  kaisi ← i , syuryo ← j
(07)  |  を実行し、そうでなくてもし soRieki < 0 ならば
(08)  |  |  soRieki ← 0 , i ← j+1
(09)  |  を実行する
(10)  を繰り返す
(11)  「開始時間帯は」と kaisi と「とし、」を表示する
(12)  「終了時間帯は」と syuryo と「とし、」を表示する
(13)  「総利益の最大値は」と saidaiRieki と「千円である。」を表示する
```

図２の行（04）にある soRieki＞saidaiRieki の比較は、すべての変数 j に対して、１回ずつ行う。
実際、$[i,\ j]$ が０以上であるとき、次に求める総利益は $[i,\ j+1]$ となり、$[i,\ j]$ が負であるときは、次の総利益は $[j+1,\ j+1]$ となるため、行（04）の比較は終了時間帯 j において１回ずつ行っていることを意味している。このことから soRieki > saidaiRieki の比較は全部で **8** 回行われる。

問３の正解

セ	4	ソ	3	タ	2	チ	0	ツ	1	テ	2
ト	8	ナ	d	ニ	2	ヌ	0	ネ	9		