

2018年度 A O 入試 1 次選考問題

情報理工学部

情報科目

注意事項

1. 試験開始の合図があるまで、問題用紙を開いてはいけません。
2. 解答はすべて、所定の解答用紙に記入してください。
3. 解答用紙に受験番号と氏名（フリガナ）を記入してください。
4. 解答時間は60分です。問題は12ページあります。
5. 問題用紙・解答用紙および計算用紙はすべて回収します。一切持ち帰ってはいけません。

[I] 次の (1) ~ (3) で示した 10 進数の数値を 8 ビット 2 進整数として表現し、それを指示した数だけビットシフトした後のビット列を解答しなさい。なお、ビットシフト操作の例を次の例 1 と例 2 で示す。

例 1 : 2 ビット左シフト シフト前 (0 0 0 0 0 0 0 1)₂

↓

シフト後 (0 0 0 0 0 1 0 0)₂

例 2 : 4 ビット右シフト シフト前 (0 0 0 1 0 0 1 0)₂

↓

シフト後 (0 0 0 0 0 0 0 1)₂

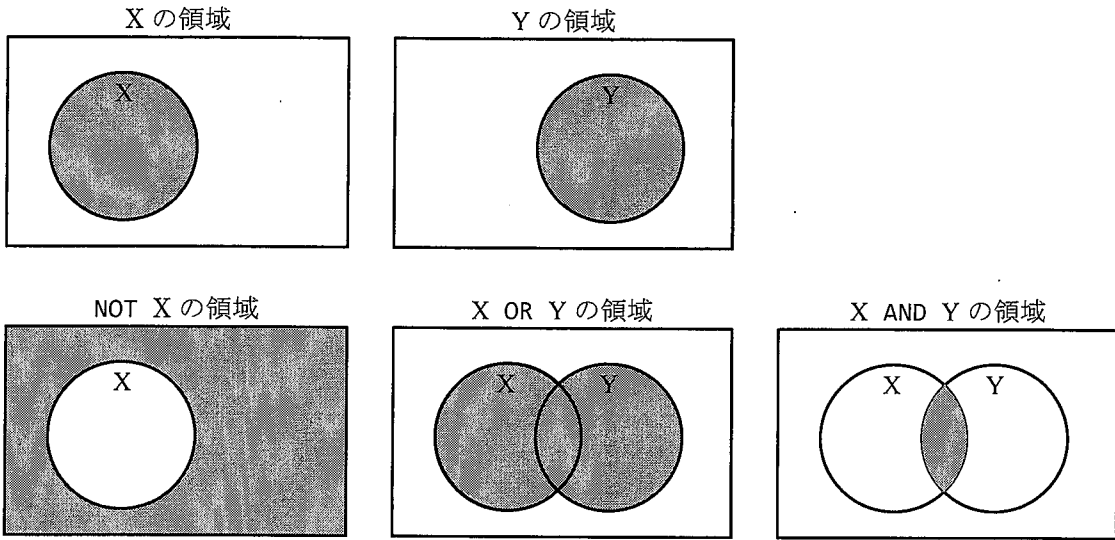
(1) (251)₁₀ → 3 ビット左シフト

(2) (100)₁₀ → 2 ビット右シフト

(3) (88)₁₀ → 3 ビット右シフト

〔II〕 以下の文章を読んで、下の設問 (A) ~ 設問 (C) に答えなさい。

以下のベン図において、条件 X や条件 Y が成り立つ領域を、円 X や円 Y の内部の灰色の領域で表すこととする。また、演算規則 NOT、OR、および AND を、それぞれ以下の灰色の領域で表すように定義する。

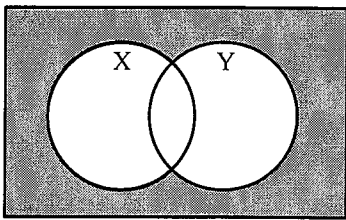


NOT、OR、AND は組み合わせて用いることができるものとし、組み合わせにおける各演算の優先順位を表すために () を用いることとする。

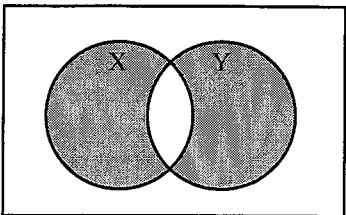
設問 (A)

以下の(i)と(ii)のベン図の灰色の領域を表す演算を、X、Y、NOT、OR、AND、() のうちの必要なものを組み合わせてそれぞれ記しなさい。

(i)



(ii)



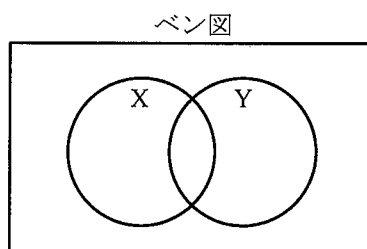
次に、プログラム内の条件文 (if 文) で上述の NOT、OR、AND、() の演算規則を用いることを考える。if 文では以下に示す命令文で表される条件分岐を用いることができ、if 文の条件式内に上述の演算規則を用いることができる。

<命令文>	<動作>
if 条件式	条件式が成り立っているとき、else までの命令を実行し、
...	成り立っていないとき、else から end までの命令を実行する。
else	
...	
end	

例として、1 から 100 までの整数値をランダムに 1 つだけ表示するルーレットの値を判定するプログラムを考える。「ルーレットの値が 50 以上であるという条件が成り立っていること」を X で表し、「ルーレットの値が偶数であるという条件が成り立っていること」を Y で表すと、ルーレットの値が 50 以上の偶数であれば「当たり」、それ以外であれば「外れ」とメッセージを出力するプログラムは次のように書ける。また、X と Y の関係を、下のベン図に表した。

[プログラム]

```
if X AND Y
    「当たり」を出力
else
    「外れ」を出力
end
```



設問 (B)

上のプログラムの else の条件に相当するベン図内の領域を、解答用紙のベン図内に塗り潰して示しなさい。

設問 (C)

上のプログラムを、AND を用いずに書く方法を次のように考えた。プログラムの残りの行を完成させなさい。

[AND を使わないプログラム (未完成)]

```
if X
    if Y
        「当たり」を出力
```

〔III〕 以下の文章を読んで、設問（A）～ 設問（C）に答えなさい。

ここでは10名の受験者のデータを加工する作業について問う。データには三科目の試験結果が含まれている。各受験者の三科目の得点のうち上位二科目について合計を求め、その上位5名を合格とする。対象とするデータを以下に示す。

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語						
2	023	山本裕太	73	52	88						
3	169	小林薫	97	55	91						
4	231	加藤美樹	79	53	85						
5	267	木村紗智子	68	100	59						
6	402	井上清美	90	44	66						
7	592	野村大輔	96	46	78						
8	667	佐古田美由	40	44	46						
9	808	今井義男	69	57	74						
10	871	坂井幸司	78	53	82						
11	907	広田かなえ	69	76	100						

このデータを加工して、合格者を決定し、その発表（掲示）に使えるデータを作成する。以下に完成したデータを示す。

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語	一位得点	二位得点	合計点			
2	169	小林薫	97	55	91	97	91	188	英	国	
3	907	広田かなえ	69	76	100	100	76	176	数	国	
4	592	野村大輔	96	46	78	96	78	174	英	国	
5	267	木村紗智子	68	100	59	100	68	168	英	数	
6	231	加藤美樹	79	53	85	85	79	164	英	国	

以降にこの完成形にたどり着くまでの経過を段階的に示す。そこで必要となる操作、設定すべき計算式などについて検討し、解答せよ。なお解答に用いることができる式や関数などは以下の表の指定に従うこと。

<p><表記></p> <p>定数 1, 2 などの整数および "A" "文字" などの文字列が書ける。</p> <p>セル番地 A-Z の列情報、1-99 の行情報で表現する。A1 などとして単一のセルを表現する。</p> <p>セル範囲 A1:B9 などと二つのセル番地を：で接続して、その二つのセルを対角位置にもつ範囲を表現する。</p> <p>= 式 式の演算結果をそのセルに表示する。式にはセル番地、四則演算子、IF文、関数および演算の優先順位を示す括弧が書ける。 例： = A2 + SUM(B2:B9)</p> <p>IF(条件式, 値1, 値2) 条件式が真の場合は値1を、偽の場合は値2を表示する。条件式は比較演算子を用いて記述する。</p>	<p><関数></p> <p>SUM(セル範囲) 指定されたセル範囲の合計値を求める。</p> <p>MAX(セル範囲) 指定されたセル範囲の最大値を求める。</p> <p>MIN(セル範囲) 指定されたセル範囲の最小値を求める。</p> <p><比較演算子></p> <p>= 両辺の式の値が等しい場合に真とする。</p> <p>!= 両辺の式の値が異なる場合に真とする。</p> <p><四則演算子></p> <p>+ 加算（足し算）を行う。</p> <p>- 減算（引き算）を行う。</p> <p>* 乗算（かけ算）を行う。</p> <p>/ 除算（割り算）を行う。</p>
---	--

step 1. 上位二科目の得点と、その合計点を求める

まず、以下のように三科目中、上位二つの得点を F, G 列に求める。その合計点も H 列に求める。

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語	一位得点	二位得点	合計点			
2	023	山本裕太	73	52	88	88	73	161			
3	169	小林薫	97	55	91	97	91	188			
4	231	加藤美樹	79	53	85	85	79	164			
5	267	木村紗智子	68	100	59	100	68	168			
6	402	井上清美	90	44	66	90	66	156			
7	592	野村大輔	96	46	78	96	78	174			
8	667	佐古田美由	40	44	46	46	44	90			
9	808	今井義男	69	57	74	74	69	143			
10	871	坂井幸司	78	53	82	82	78	160			
11	907	広田かなえ	69	76	100	100	76	176			

設問 (A)

F2, G2, H2 セルに設定すべき式を示せ。

step 2. 合計点が高い受験者の順に並べ替えて以下のようにする

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語	一位得点	二位得点	合計点			
2	169	小林薫	97	55	91	97	91	188			
3	907	広田かなえ	69	76	100	100	76	176			
4	592	野村大輔	96	46	78	96	78	174			
5	267	木村紗智子	68	100	59	100	68	168			
6	231	加藤美樹	79	53	85	85	79	164			
7	023	山本裕太	73	52	88	88	73	161			
8	871	坂井幸司	78	53	82	82	78	160			
9	402	井上清美	90	44	66	90	66	156			
10	808	今井義男	69	57	74	74	69	143			
11	667	佐古田美由	40	44	46	46	44	90			

設問 (B)

並べ替え (ソート処理) の操作で示すべき内容 (ソート範囲・ソートキー・並び順) を示せ。

step 3. 不合格者の削除

上位 5 名を残して、他のデータを削除して以下のようにする。

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語	一位得点	二位得点	合計点			
2	169	小林薫	97	55	91	97	91	188			
3	907	広田かなえ	69	76	100	100	76	176			
4	592	野村大輔	96	46	78	96	78	174			
5	267	木村紗智子	68	100	59	100	68	168			
6	231	加藤美樹	79	53	85	85	79	164			

step 4. 科目ラベルの作成

合格発表では、学生番号だけでなく、得点計算のために用いられた上位二科目がどの科目だったのかを示したい。

例えば英語と国語が対象二科目となった場合は、以下のように「英」「国」を表示する。ただし二科目あるいは三科目の得点と同じであることは無いものとする。

	A	B	C	D	E	F	G	H	I	J	K
1	学生番号	氏名	英語	数学	国語	一位得点	二位得点	合計点			
2	169	小林薫	97	55	91	97	91	188	英		国
3	907	広田かなえ	69	76	100	100	76	176		数	国
4	592	野村大輔	96	46	78	96	78	174	英		国
5	267	木村紗智子	68	100	59	100	68	168	英	数	
6	231	加藤美樹	79	53	85	85	79	164	英		国

設問 (C)

I2セルに設定すべき式を示せ。

設問 C の解答と同様の式を I, J, K 列の必要なセルに設定することで、データは完成する。

[IV] 以下の文章を読んで、設問 (A) ~ 設問 (C) に答えなさい。

気温を観測するためにコンピュータに温度センサを接続し、5分に一回、図1に示すプログラムを動作させて気温のデータを記録している。気温データは変数 `t` に読み出された後、配列 `buf` に格納される。配列 `buf` は要素数が10で、配列の添字は0から9までを使用できる。また、変数 `index` は初期値0で、このコンピュータの動作中は変数 `index` と配列 `buf` は格納された値を保持し続けているものとする。

コンピュータが動作を開始してから十分な時間が経過した後、配列 `buf` には最新の10個のデータが格納されていることになる。このようなデータ構造はリングバッファと呼ばれる。

```
t = get_temp() /* 気温を取り出して t に代入 */
buf[index] = t
index = index + 1
if (index >= 10)
    index = 0
end
```

図1 気温データを配列に格納するプログラム

この問 [IV] および次の問 [V] では、次ページに示す記法を使ってプログラムを記述する。

設問 (A)

図1のプログラムの後ろに記述を追加して、変数 `highest` が、観測を開始してからこれまでの最高気温を保持するようにしたい。追加すべき適切なプログラム行を記述しなさい。

なお、変数 `highest` はこのコンピュータの動作中は値を保持し続け、初期値には `-999` などの、気温としてはあり得ない低い値が設定されていると考えてよい。

プログラムの記法の説明

<p><命令文> 変数 = 式</p>	<p><動作> 変数に式の値を代入する。以下の説明ではこの文を「代入文」と呼ぶ。</p>
<p>for (代入文 1 ; 条件式 ; 代入文 2) ... end</p>	<p>まず、代入文 1 を実行し、条件式を評価する。条件式が偽であれば何もしない。条件式が真のとき、end までの命令を実行し、次に代入文 2 を実行する。その後、条件式が真である間、end までの命令と代入文 2 を実行する。</p>
<p>if 条件式 ... else ... end</p>	<p>条件式が真のとき、else までの命令を実行し、偽のとき、else から end までの命令を実行する。else を記述しない場合、条件式が真のときに end までの命令を実行し、条件式が偽のときは何もしない。</p>
<p><関数> func F (x) ... end</p>	<p><説明> 名前Fを持つ関数（サブルーチン）を定義する。xは引数で、実行の際に呼び出し側から値が渡される。引数を指定しない場合は（ ）だけを記述し、複数の引数を指定する場合は「,」で区切る。</p>
<p>F (式)</p>	<p>名前Fを持つ関数（サブルーチン）を実行する。式の値は引数として関数に渡される。式は、関数の定義で示された引数の数だけ記述する。引数が必要ない場合は（ ）だけを記述し、複数の引数がある場合は、foo(3, a+1) のように「,」で区切る。</p>
<p><比較演算子> A == B A != B A <= B A < B A >= B A > B</p>	<p><意味> AとBの値が等しい。 AとBの値が等しくない。 Aの値がBの値以下である。 Aの値がBの値より小さい。 Aの値がBの値以上である。 Aの値がBの値より大きい。</p>
<p><算術演算子> + - * /</p>	<p><意味> 加算（足し算）を行う。 減算（引き算）を行う。 乗算（掛け算）を行う。 除算（割り算）を行う。ただし、結果は実数で表される。</p>
<p>複数の算術演算子が混在した式では * と / の計算が + や - よりも先に行われる。また、式の中で（ ）を使い、計算の順序を示すことができる。</p>	

設問 (B)

図1のプログラムを繰り返し呼び出して観測を続けている間、別のプログラムを動作させて、配列 buf に格納されたデータを参照して利用することができる。図2は最も新しい観測データを変数 current に格納するプログラムであり、空欄部分では変数 x の値を計算している。空欄を埋める適切な記述を答えなさい。

```
x = index
```



```
current = buf[x]
```

図2 最も新しい観測データを変数 current に格納する

設問 (C)

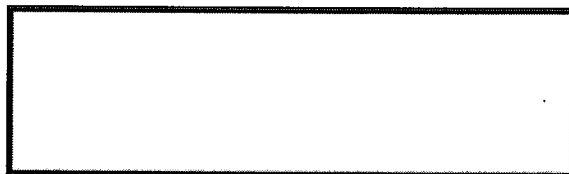
最近数十分程度の気温の傾向を知るため、最近の観測データほど重要視して加重平均（重み付き平均）を計算したい。最も新しい観測データを d_0 、前回のデータを d_1 、その前と、さらにその前を d_2 、 d_3 としたとき、次の式のように加重平均 avg を求める。

$$\text{avg} = (d_0 * 4 + d_1 * 2 + d_2 + d_3) / 8$$

いま、要素数4の配列 w に、4, 2, 1, 1 という順序で値が格納されているとする。また、変数として sum, i, x, avg が利用できるとする。配列 buf には10個の有効なデータが格納されていると仮定してよい。

図3の空欄を埋める適切な記述を答えなさい。

```
sum = 0  
x = index  
for (i = 0; i < 4; i = i + 1)
```



```
end  
avg = sum / 8
```

図3 最新の4つの観測データから加重平均を計算する

[V] 以下の文章を読んで、設問 (A) ~ 設問 (C) に答えなさい。なお、この問いでは、前問 [IV] で示した記法に従ってプログラムを記述するものとする。

N を正の整数とし、 $N \times N$ の 2 次元配列を考える。2 次元配列 a の i 行 j 列の要素を表すために $a[i, j]$ という記法を用いる (図 1 は 4×4 の 2 次元配列の例)。

たとえば、 4×4 の配列 a の 0 行目のすべての要素の値を表示するには図 2 のようなプログラムを記述すればよい。ただし、`print` という関数を使って値が表示でき、変数として i が利用できるものとする。

	0	1	2	3
0	[0,0]	[0,1]	[0,2]	[0,3]
1	[1,0]	[1,1]	[1,2]	[1,3]
2	[2,0]	[2,1]	[2,2]	[2,3]
3	[3,0]	[3,1]	[3,2]	[3,3]

図 1 4×4 の配列の各要素の表記

	0	1	2	3
0	→			
1				
2				
3				

(a) 表示する要素と順番

```
for (i = 0; i < 4; i = i + 1)
    print( a[0, i] )
end
```

(b) プログラム例

図 2 4×4 の配列の 0 行目を表示する

設問 (A)

図 3 の矢印に示すような順序で 4×4 の配列 a の外周の要素 (12 個) の値を表示するプログラムを記述しなさい。変数として i, j を利用してよい (必ずしもすべてを利用する必要はない)。

	0	1	2	3
0	→			
1				
2				
3	←			

図 3 4×4 の配列の外周を表示する

次に、関数（サブルーチン）を定義して利用する例を示す。

図4は、 4×4 の2次元配列 a の m 行目 ($0 \leq m < 4$) の要素を表示する関数 `showline` の記述で、 m はこの関数の引数である。たとえば、配列 a の1行目の要素を表示させるには、`showline(1)` という呼び出しをプログラムに記述すればよい。

```
func showline(m)
  for (i = 0; i < 4; i = i + 1)
    print( a[m, i] )
  end
end
```

図4 4×4 の配列の m 行目を表示する関数 `showline`

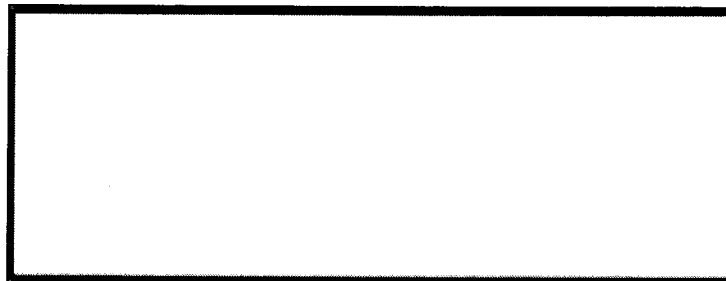
設問 (B)

図5は配列 a の要素 $a[1, 1]$ から始まる、一辺の長さが4の正方形の経路（要素の表示順）を示す。

このように、要素 $a[s, s]$ から始まり、一辺の長さが e である正方形の経路の順序で配列の要素を表示する関数 `path` を記述したい。下の記述で、 s, e は関数の引数である。空欄を埋めて関数の記述を完成させなさい。なお、 $s \geq 0$ 、 $e \geq 2$ と仮定してよい。変数として i, j, k が利用できる（必ずしもすべてを利用する必要はない）。

この関数を使えば、図3、図5の例は、それぞれ、`path(0, 4)` および `path(1, 4)` のような呼び出しで実現することができる。

```
func path(s, e)
```



```
end
```

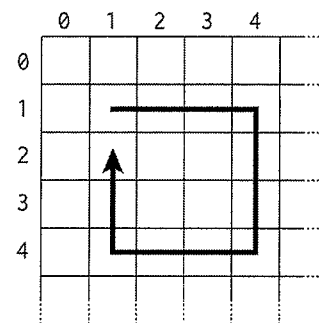


図5 正方形の経路の例

設問 (C)

設問 (B) で定義した関数 `path` を使うと、図 6 のように、 $N \times N$ の 2 次元配列 `a` を渦巻き状にたどって要素を表示させることができる。

いま、配列 `a` が 8×8 の大きさであると仮定する。for 文を使った繰り返し処理の中で関数 `path` を実行し、配列 `a` のすべての要素を渦巻き状にたどって表示するプログラムを記述しなさい。変数として `i`, `j`, `k` を利用してよい (必ずしもすべてを利用する必要はない)。

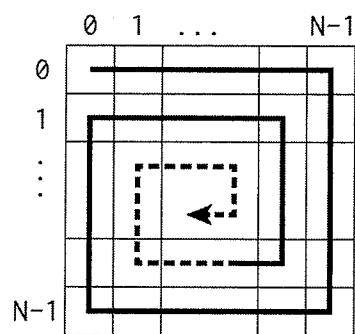


図 6 渦巻き状の経路