

2020 年度
慶應義塾大学入学試験問題

環境情報学部

数学または情報

注意事項 1

1. 試験開始の合図があるまで、この問題冊子を開かないでください。
2. この冊子は全部で28ページです。
 - ・数学の問題Ⅰ～Ⅵは3ページから13ページです。
 - ・情報の問題Ⅰ～Ⅴは14ページから28ページです。

試験開始の合図とともにすべてのページが揃っているか確認してください。

ページの欠落・重複があった場合には、直ちに監督者に申し出てください。

3. 問題冊子の2ページに「注意事項2」があります。試験開始後必ず読んでください。
4. 数学・情報のいずれか1つを選択し、解答用紙の選択科目名の欄に科目名を記入し、選択科目マーク欄にマークしてください。
5. 問題冊子は、試験終了後必ず持ち帰ってください。
6. 受験番号と氏名は、解答用紙の所定の欄に必ず記入してください。
7. 解答用紙の「注意事項」を必ず読んでください。

注意事項 2

問題冊子に数字の入った があります。それらの数字は解答用紙の解答欄の番号を表しています。対応する番号の解答欄の 0 から 9 までの数字または - (マイナスの符号) をマークしてください。

が 2 個以上つながったとき、数は右詰めで入れ、左の余った空欄には 0 を入れてください。負の数の場合には、マイナスの符号を先頭の に入れてください。

$$(例) \quad 12 \rightarrow \boxed{0} \boxed{1} \boxed{2}$$

$$-3 \rightarrow \boxed{-} \boxed{0} \boxed{3}$$

分数は約分した形で解答してください。マイナスの符号は分母には使えません。

$$\begin{aligned} (例) \quad \frac{4}{8} &\rightarrow \frac{1}{2} \rightarrow \frac{\boxed{0} \boxed{1}}{\boxed{0} \boxed{2}} \\ -\frac{6}{9} &\rightarrow -\frac{2}{3} \rightarrow \frac{\boxed{-} \boxed{2}}{\boxed{0} \boxed{3}} \end{aligned}$$

ルート記号の中は平方因子を含まない形で解答してください。

$$(例) \quad \sqrt{50} \rightarrow \boxed{0} \boxed{5} \sqrt{\boxed{0} \boxed{2}}$$

$$-\sqrt{24} \rightarrow \boxed{-} \boxed{2} \sqrt{\boxed{0} \boxed{6}}$$

$$\sqrt{13} \rightarrow \boxed{0} \boxed{1} \sqrt{\boxed{1} \boxed{3}}$$

数式については、つぎの例のようにしてください。分数式は約分した形で解答してください。

$$(例) \quad -a^2 - 5 \rightarrow \boxed{-} \boxed{1} a^2 + \boxed{0} \boxed{0} a + \boxed{-} \boxed{5}$$

$$\frac{4a}{2a - 2} \rightarrow \frac{-2a}{1 - a} \rightarrow \frac{\boxed{0} \boxed{0} + \boxed{-} \boxed{2} a}{1 - \boxed{0} \boxed{1} a}$$

選択肢の番号を選ぶ問題では、最も適切な選択肢を 1 つだけ選びなさい。また、同じ選択肢を何回選んでもかまいません。

情報 I

以下、法制度に関しては、日本のものについて考えるものとする。

(ア) 著作権法に関する説明として、正しいものを次の選択肢から選び、その番号を解答欄 (1) にマークしなさい。

- (1) 書店で購入した小説を公の場で朗読することは、著作権の侵害にあたらない。
- (2) 既存の著作物に依拠せず、独自に創作した作品でも、結果として表現が類似している場合には、著作権の侵害となる。
- (3) 著作権法上、複製とは、印刷、写真、複写、録音、録画その他の方法により有形的に再製することをいう。
- (4) 著作権を侵害する書籍が販売されている場合でも、出版の差止めは認められないが、著作権者は損害賠償を請求することができる。
- (5) 著作権を侵害する行為には、刑事罰は定められていない。

(イ) 次の文章を読み、空欄 (2) から (5) にあてはまるものを選択肢から選び、その番号をそれぞれの解答欄にマークしなさい。

文化は先人の文化的所産を利用しながら発展し、新しい著作物や作品を創作する際に、他人の著作物を採録することが必要な場合はよくある。とりわけ、報道・批評等の場合には (2) の自由を保障する意味でも (3) の必要性が高い。そこで、著作権法 32 条は、 (4) された著作物について、 (5) な慣行に合致し、かつ報道等のその他の (3) の目的上正当な範囲内で行われる (5) の場合には利用できると、抽象的な要件を規定している。

(出典：駒田泰土・潮海久雄・山根崇邦『知的財産法Ⅱ 著作権法』(有斐閣、2016 年)

(2)～ (5) の選択肢】

- (1) 平等 (2) 公表 (3) 流用 (4) 信仰 (5) 隠匿
- (6) 引用 (7) 表現 (8) 営業 (9) 公正 (0) 私的

(ウ) 特許法に関する説明として、正しいものを次の選択肢から選び、その番号を解答欄 (6) にマークしなさい。

- (1) 特許出願した発明の内容は、特許庁の審査により特許権が認められない場合には公開されない。
- (2) 特許出願前に公然と知られていた発明であっても、自分で技術を発明した者であれば、他の人が特許出願をしていない限り、特許権を取得することができる。
- (3) 当該技術分野の平均的な知識を有する技術者が、既存の技術に基づいて容易に考えられる発明については、特許権は取得できない。
- (4) 特許権の技術的範囲に含まれる製品は、特許権者から適法に購入したものであっても、特許権者の承諾なく転売することができない。

(5) 医薬品を製造するための技術は、人道的な見地から、特許権により独占することはできない。

(エ) 個人情報保護法（個人情報の保護に関する法律）に関する説明として、正しいものを次の選択肢から選び、その番号を解答欄 (7) にマークしなさい。

- (1) 個人情報取扱事業者は、適法に取得して利用目的の達成に必要な範囲内で利用している個人データについても、本人から利用の停止又は消去の請求を受けたときは、これに応じなければならない。
- (2) 個人情報取扱事業者は、本人の求めに応じて個人データの第三者への提供を停止することとしている場合、届出や通知等の措置をとることなく、当該個人データを第三者に提供することができる。
- (3) 生存する個人に関する情報ではなく、もっぱら死亡した個人のみに関する情報も、個人情報に該当する。
- (4) 情報に含まれる氏名、生年月日その他の記述等により特定の個人を識別することができない場合、他の情報と容易に照合することができ、それにより特定の個人を識別することができるとしても、その情報は個人情報に該当しない。
- (5) 個人情報取扱事業者は、本人の同意なく個人情報の利用目的を変更する場合には、変更前の利用目的と関連性を有すると合理的に認められる範囲を超える変更をしてはならない。

(オ) プロバイダ責任制限法（特定電気通信役務提供者の損害賠償責任の制限及び発信者情報の開示に関する法律）が定めるルールの説明として、誤っているものを次の選択肢から選び、その番号を解答欄 (8) にマークしなさい。

- (1) プロバイダが、第三者の名誉を毀損する内容のファイルの発信者による掲載を放置している場合に、裁判所がプロバイダにファイルの削除を命令するときのルール。
- (2) プロバイダが、第三者からの著作権侵害の主張を信じて記事を削除した場合に、実際には著作権を侵害していなかった発信者がプロバイダに損害賠償を請求したときのルール。
- (3) プロバイダが、第三者からの名誉毀損の主張を受けて発信者に削除の可否を照会したにもかかわらず、発信者が照会を無視したため記事を削除したプロバイダに、発信者が損害賠償を請求したときのルール。
- (4) プロバイダが、第三者から名誉毀損に基づく発信者情報の開示請求を受けたものの、違法性の判断がつかないため情報を開示しなかった場合に、後日裁判所で名誉毀損の被害が認められた第三者がプロバイダに損害賠償を請求したときのルール。
- (5) プロバイダが、第三者から著作権侵害に基づく発信者情報の開示請求を受けた場合に、開示の可否について発信者の意見を聞くときのルール。

情報II

古くから知られている川渡のパズルについて説明した次の文章を読み、空欄 (9) (10) から (17) (18) に入る数字をそれぞれの解答欄にマークしなさい。

川渡のパズルとは、たとえば、川の手前側の岸 (A) に、人 (n)、狼 (w)、キャベツ (c)、ヤギ (g) が存在し、ボートに乗って対岸 (B) に、ある制約条件の下で全部が渡るには、どのような順番でボートに乗つて渡ればいいかを考える問題である。

ここでは、次のような制約条件の元でパズルを解くことを考える。

制約条件 1：人がいない状況でヤギと狼が片方の岸に揃うとヤギが狼に食べられてしまうので、そのような状況になってはいけない。

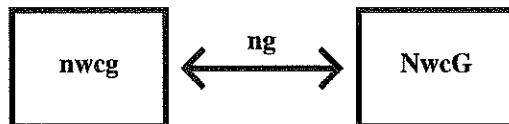
制約条件 2：人がいない状況でヤギとキャベツが片方の岸に揃うとヤギがキャベツを食べてしまうので、そのような状況になってはいけない。

制約条件 3：ボートには、人だけ乗って対岸に移動するか、人の他に一つだけ、つまり、狼、キャベツ、ヤギのうちの一つと一緒に乗って対岸に移動するしかできない。

状態を簡単に表現するために、A に存在するものは英小文字で、B に存在するものを英大文字で表すことにすると、4 つ全部が A に存在することは、nwcg と表せ、全部が B に渡った状況を NWCG と表わせる。大文字小文字が混ざった 4 つの文字は、そのいずれでもない状態を表すことになる。

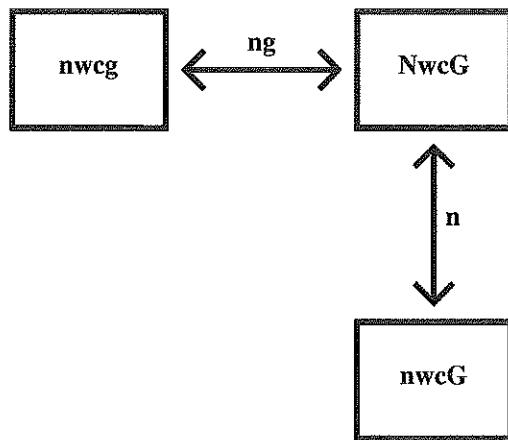
上記の制約条件から、Nwcg のような到達してはいけない状態にならずに、nwcg という状態から NWCG という状態に変化させる手順を探すことが、このパズルを解くことになる。

パズルを解くために、制約条件を満たす状態と、ある状態から別の状態へどのようにボートを使って変化させられるかを図で表することにする。最初の状態、nwcg から、人がヤギと一緒に川を渡った場合の状態の変化を示す図は状態とそれを結ぶ矢印で表される。



渡ったのと同じように川を渡って戻れば、変化する前の状態に戻るので、矢印は双方向になっていて、移動する人とヤギを表す文字が矢印に付加されている。

さらに移動して得られる状態を一つ付け加えると次のような、状態が 3 個になった図が得られる。



最初の状態 (nwcg) から、制約条件を満たすように移動して到達できるすべての状態を考え、それらの状態間の変化を示す図を完成させると、パズルが解け、ある状態から別の状態に変化するのに、必要なポートでの移動の最小回数が分かる。

図を完成させたとき、図中の状態の数は、(9) (10) 個になる。nwcg から NWCG になるためには、最低でも (11) (12) 回の状態の変化、つまり、この回数だけポートによる移動が必要なことが分かる。また、この最低の変化数で済む状態変化の手順は (13) (14) 通りになる。

次に、この完成された図の各状態につながっている矢印の数を考える。矢印が一つの状態は (15) (16) 個あり、矢印が二つの状態は (17) (18) 個ある。

情報III

(ア) $y = h(x)$ を、任意のビット列 x を引数にとり、16 ビットの値(ハッシュ値) y を返すハッシュ関数とする。ただし、 y を与えて x を求める方法は知られていない。

(a) 任意のビット列 a と 20 ビットの値 b から成るデータにおいて、 a はそのままに b だけを変化させて、ハッシュ値 $y_1 = h(concat(a, b))$ を任意の値にすることを考える。ここで、関数 $concat$ は 2 つのビット列を連結する関数であり、例えは $concat(001111, 00)$ は 00111100 となる。

ハッシュ関数の特性から、 y_1 からハッシュ関数に渡す引数 $concat(a, b)$ を計算することはできず、そのため b も計算することはできない。よってハッシュ値が y_1 となるような b を選ぶためには、任意の b を選んでハッシュ値を計算する作業を、計算したハッシュ値が y_1 になるまで繰り返す必要がある。ハッシュ関数 $y = h(x)$ は、 x から y を計算したときに y の出現確率が同じでなければならないとする。この方法でハッシュ値が y_1 になるように b を選ぶのに必要なハッシュ値の計算回数の期待値は

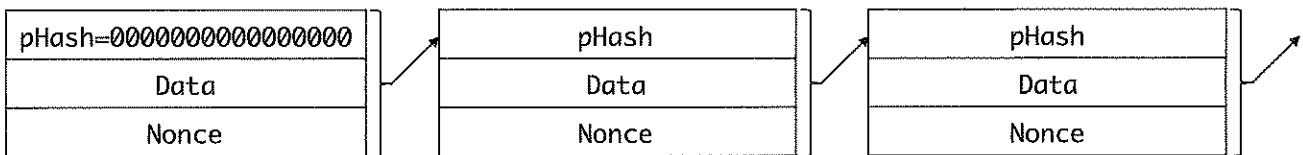
$(19) \quad (20) \quad (21) \quad (22) \quad (23) \quad (24) \quad (25)$ 回である。これは、次のようにして求めることができる。

$$\frac{1}{2} \cdot 2^{(26) \quad (27)}$$

また、ハッシュ値 y_2 の上位 8 ビットが 0 になるような b を選ぶのに必要なハッシュ値の計算回数の期待値は $(28) \quad (29) \quad (30) \quad (31) \quad (32) \quad (33) \quad (34)$ 回である。ただし、 y_2 の下位 8 ビットは何でもよいものとする。これは、次のようにして求めることができる。

$$\frac{1}{2} \cdot 2^{(35) \quad (36)}$$

(イ) 16 ビットの値 pHash、任意のビット列 Data、32 ビットの値 Nonce から成るデータセットを考え、pHash, Data, Nonce を連結したビット列をブロックと呼ぶこととする。ここで、ハッシュ関数 h を用いて下図のように pHash に前のブロックのハッシュ値を入れることによってブロックのチェーンを作ることを考える。ただし、チェーンの 1 つ目のブロックの pHash は 0 が 16 個並んだものとする。また、それぞれのブロックのハッシュ値の上位 8 ビットは 0 でなければならないものとする。ただし、ハッシュ値の下位 8 ビットは何でもよいものとする。



(a) 1 つ目のブロックが外部から与えられたとする。2 つ目のブロックの Data が決まっている状態で、2 つ目のブロックを生成するために 2 つ目のブロックの Nonce を選ぶのに必要なハッシュ値の計算回数

の期待値は $\boxed{(37)} \boxed{(38)} \boxed{(39)} \boxed{(40)} \boxed{(41)} \boxed{(42)} \boxed{(43)}$ 回である。これは、次のようにして求めることができる。

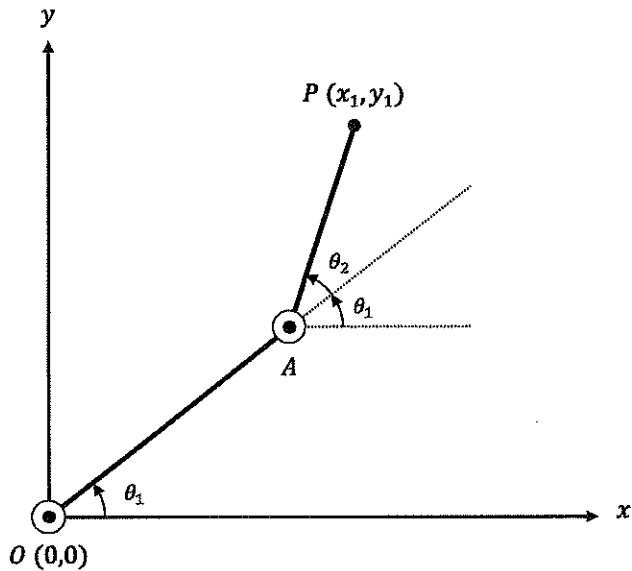
$$\frac{1}{2} \cdot 2 \left[\begin{array}{|c|c|} \hline (44) & (45) \\ \hline \end{array} \right] + \left[\begin{array}{|c|c|} \hline (46) & (47) \\ \hline \end{array} \right]$$

(b) 10 個のブロックが外部から与えられたとき、3 つ目のブロックの Data を変更することを考える。10 個のブロック全ての辻縁が合うようにするために必要なハッシュ値の計算回数の期待値は $\boxed{(48)} \boxed{(49)} \boxed{(50)} \boxed{(51)} \boxed{(52)} \boxed{(53)} \boxed{(54)} \boxed{(55)}$ 回である。これは、次のようにして求めることができる。

$$\frac{1}{2} \cdot 2 \left[\begin{array}{|c|c|} \hline (44) & (45) \\ \hline \end{array} \right] \cdot \left[\begin{array}{|c|c|} \hline (56) & (57) \\ \hline \end{array} \right] + \left[\begin{array}{|c|c|} \hline (58) & (59) \\ \hline \end{array} \right]$$

情報IV

図のような 2 つの関節 O と A を持つロボットアームを制御する方法を考える。次の文章の空欄
 $\boxed{(60)} \boxed{(61)}$ から $\boxed{(78)} \boxed{(79)}$ にはもっとも適したものを選択肢から選び、解答欄にマークしなさい。また、空欄
 $\boxed{(80)} \boxed{(81)} \boxed{(82)}$ および $\boxed{(83)} \boxed{(84)} \boxed{(85)}$ には適切な数字を解答欄にマークしなさい。ただし、このロボットアーム
 は全ての関節および腕が $x \geq 0, y \geq 0$ の範囲で動作するものとする。



(ア) 各関節の関節角度を θ_1 および θ_2 、2 つの腕の長さ $\overline{OA}, \overline{AP}$ をそれぞれ L_1, L_2 とすると、手先の位置 $P(x_1, y_1)$ は次の式で求められる。

$$x_1 = L_1 \boxed{(60)} \boxed{(61)} + L_2 \boxed{(62)} \boxed{(63)}$$

$$y_1 = L_1 \boxed{(64)} \boxed{(65)} + L_2 \boxed{(66)} \boxed{(67)}$$

(イ) 次にロボットアームを動かして座標 (x_1, y_1) にあるコーヒーカップを掴む状況を考える。この場合は逆に、移動させたい手先の位置 (x_1, y_1) から、各関節に指示する関節角度 (θ_1, θ_2) を計算しなければならない。図のように、座標原点を O 、手先位置を $P(x_1, y_1)$ とすると、各関節角度は次式の様に表すことができる。ただし、 $\cos \alpha = \frac{x_1}{OP}$ とする。また、各関節の可動範囲は、 $0^\circ \leq \theta_1 \leq 90^\circ$ 、 $0^\circ \leq \theta_2 \leq 180^\circ$ であり、座標 (x_1, y_1) はロボットアームの可動範囲で指定されるものとする。

$$\theta_1 = \alpha - \boxed{(68)} \boxed{(69)}$$

$$\theta_2 = 180^\circ - \boxed{(70)} \boxed{(71)}$$

ここで、三角形 OAP に関して、 \overline{OP} と $\angle PAO$ および $\angle POA$ の間には、次式に示す関係が成立する。

$$\cos \angle PAO = \frac{\boxed{(72)} \boxed{(73)}}{\boxed{(74)} \boxed{(75)}}$$

$$\cos \angle POA = \frac{\boxed{(76)} \boxed{(77)}}{\boxed{(78)} \boxed{(79)}}$$

いま、腕の長さがそれぞれ $L_1 = \sqrt{3}$ 、 $L_2 = 1$ であるとき、手先位置 $P(x_1, y_1)$ を $(0, 2)$ に移動させるために必要となる関節角度は、それぞれ次のようになる。

$$\theta_1 = \boxed{(80)} \boxed{(81)} \boxed{(82)}^\circ$$

$$\theta_2 = \boxed{(83)} \boxed{(84)} \boxed{(85)}^\circ$$

【 $\boxed{(60)} \boxed{(61)} \sim \boxed{(78)} \boxed{(79)}$ の選択肢】

- | | | | |
|--|--|--|--|
| (11) $\angle PAO$ | (12) $\angle POA$ | (13) $\angle OPA$ | (14) θ_1 |
| (15) θ_2 | (16) L_1 | (17) L_2 | (18) $L_1 L_2$ |
| (19) $2 L_1 L_2$ | (20) $3 L_1 L_2$ | (21) $4 L_1 L_2$ | (22) $L_1 \overline{OP}$ |
| (23) $2 L_1 \overline{OP}$ | (24) $3 L_1 \overline{OP}$ | (25) $4 L_1 \overline{OP}$ | (26) $L_2 \overline{OP}$ |
| (27) $2 L_2 \overline{OP}$ | (28) $3 L_2 \overline{OP}$ | (29) $4 L_2 \overline{OP}$ | (30) L_1^2 |
| (31) L_2^2 | (32) \overline{OP}^2 | (33) $L_1 + L_2$ | (34) $L_1 + \overline{OP}$ |
| (35) $L_2 + \overline{OP}$ | (36) $L_1^2 + L_2^2$ | (37) $L_1^2 + \overline{OP}^2$ | (38) $L_2^2 + \overline{OP}^2$ |
| (39) $L_1^2 + L_2^2 + \overline{OP}^2$ | (40) $L_1^2 - L_2^2 + \overline{OP}^2$ | (41) $L_1^2 + L_2^2 - \overline{OP}^2$ | (42) $L_2^2 - L_1^2 + \overline{OP}^2$ |
| (43) $\cos \theta_1$ | (44) $\cos \theta_2$ | (45) $\sin \theta_1$ | (46) $\sin \theta_2$ |
| (47) $\cos(\theta_1 + \theta_2)$ | (48) $\cos(\theta_1 - \theta_2)$ | (49) $\cos(\theta_2 - \theta_1)$ | (50) $\sin(\theta_1 + \theta_2)$ |
| (51) $\sin(\theta_1 - \theta_2)$ | (51) $\sin(\theta_2 - \theta_1)$ | | |

情報V

互いに異なる n 個の数が与えられている。ただし $n \geq 2$ である。以後、「順列」とは、与えられた数を並べ替えた数列を意味するものとする。

順列 P, Q に対して、辞書順で P が Q より前に来るなどを $P < Q$ と書くことにする。ただし辞書順とは、2つの順列 p_1, p_2, \dots, p_n と q_1, q_2, \dots, q_n の順序を決める際、まず p_1 と q_1 を比較し、もし $p_1 = q_1$ なら p_2 と q_2 を比較し、もし $p_2 = q_2$ なら p_3 と q_3 を比較し、と順に比較していく、最初に異なる数が出てきた時に小さい方の順列を前にするという方式である。例えば、1,5,8,4 が与えられている場合は $(1, 4, 5, 8) < (1, 4, 8, 5) < (1, 5, 4, 8) < (1, 5, 8, 4) < (1, 8, 4, 5) < \cdots < (8, 5, 1, 4) < (8, 5, 4, 1)$ となる。

(ア) 順列 a_1, a_2, \dots, a_n に対して、辞書順でそのすぐ後に来る順列を求める（例えば、上の例で 1,5,8,4 に対して 1,8,4,5 と答える）には、次のようにすればよいことが知られている。

1. 順列の末尾を含む、降順に並んでいる部分を R とする（1,5,8,4 が与えられた時は 8,4 の部分）。もし、それが順列全体なら、この順列が辞書順で最後である。
2. R のすぐ左の数を x とする（1,5,8,4 が与えられた時は 5）。
3. 順列の末尾から順に探して初めて現れる x より大きい数を y とする（1,5,8,4 が与えられた時は 8）。
4. x と y を入れ替える。
5. 入れ替えた後の R を逆順に並べ替える（1,5,8,4 が与えられた時は、上の入れ替えで 1,8,5,4 になり、 R の位置にある 5,4 を逆順にして 1,8,4,5）。

この手順をアルゴリズムの形で書くと次のようになる。空欄に入るもっとも適切なものを下の選択肢から選び、その番号をマークしなさい。

変数 a_1, a_2, \dots, a_n の値は、入力された順列とする

変数 i の値を n にする

条件 $\boxed{\text{ (86) }}$ が成り立つ間、次の処理 A を繰り返す

処理 A の始め

i の値を 1 減らす

もし $i = 1$ ならば、「辞書順で最後です」と出力してアルゴリズムを終了する

処理 A の終わり

変数 j の値を n にする

条件 $\boxed{\text{ (87) }}$ が成り立つ間、次の処理 B を繰り返す

処理 B の始め

j の値を 1 減らす

処理 B の終わり

a_{i-1} と a_j の値を交換する

j の値を n にする

条件 $\boxed{\text{ (88) }}$ が成り立つ間、次の処理 C を繰り返す

処理 C の始め

a_i と a_j の値を交換する

i の値を 1 増やす

j の値を 1 減らす

処理 C の終わり

a_1, a_2, \dots, a_n の値を出力する

【 $\boxed{\text{ (86) }} \sim \boxed{\text{ (88) }}$ の選択肢】

- (1) $a_i \geq 1$
- (2) $a_{i-1} \geq 1$
- (3) $a_{i-1} \geq a_i$
- (4) $a_j \geq a_i$
- (5) $a_j \leq a_{i-1}$
- (6) $a_j \leq a_n$
- (7) $i > 1$
- (8) $j > 1$
- (9) $i < j$

(イ) 上のアルゴリズムが正しいことの証明は次のようにしてできる。空欄に入るもっとも適切なものを下の選択肢から選び、その番号をマークしなさい。

補題: 互いに異なる m 個の数が与えられた時、辞書順で最初の順列は与えられた数を昇順に並び替えたもの、最後の順列は降順に並び替えたものである。

補題の証明: 昇順に並び替えたものを b_1, \dots, b_m とする。これより辞書順で先に来る順列があると仮定し、それを c_1, \dots, c_m とする。辞書順の定義から、ある i があって、すべての $j < i$ について $b_j = \boxed{(89)} \boxed{(90)}$ 、 $b_i > c_i$ となる。

- $i = 1$ の場合、 b_1, b_2, \dots, b_m は昇順に並んでいるから、 $b_1 > c_1$ は順列が与えられた数を並べ替えたものという仮定に反する。
- $i > 1$ の場合、順列は与えられた数を並べ替えたものであり、 b_1, b_2, \dots, b_m は昇順に並んでいるから、 $b_i > c_i$ ということは、ある $k < i$ について $c_i = b_k$ が成り立つ。ところが、すべての $j < i$ について $b_j = \boxed{(89)} \boxed{(90)}$ なので、 $c_i = b_k = \boxed{(91)} \boxed{(92)}$ となる。これは与えられた数が互いに異なるという仮定に反する。

したがって、そのような c_1, \dots, c_m は存在せず、 b_1, b_2, \dots, b_m は辞書順で最初である。降順に並び替えたものも同様に証明できる。

アルゴリズムの証明: a_n を含んで降順に並んでいる部分を a_i, \dots, a_n とする。もし $i = 1$ ならば、補題よりこれが辞書順で最後である。以後は $i > 1$ とする。

a_i, \dots, a_n が降順で、 a_{i-1}, \dots, a_n は降順にならないことから $a_{i-1} < a_i$ となる。したがって a_i, \dots, a_n の中に必ず a_{i-1} より大きい数が存在する。その中で最も右にあるものを a_j とする。

辞書順で A の次の順列を $B = (b_1, \dots, b_n)$ とする。また、順列 $C = (c_1, \dots, c_n)$ を、すべての $k \leq i-2$ に対して $c_k = a_k$ 、 $c_{i-1} = a_j$ である適当な順列とする。 $a_{i-1} < a_j = c_{i-1}$ であるから、辞書順の定義より $A < C$ となる。

まず、すべての $k \leq i-2$ に対して $b_k = a_k$ であることを示す。そうでないと仮定して、 $b_k \neq a_k$ となる k の中で最小のものを k' とする。 $A < B$ であるから、 $a_{k'} < b_{k'}$ となる。ところが、 $k' \leq i-2$ かつ $\boxed{(93)} \boxed{(94)} = a_{k'} < b_{k'}$ なので $C < B$ 、したがって $A < C < B$ となり、 B が辞書順で A の次の順列という仮定に反する。

次に、 $b_{i-1} = a_j$ であることを示す。そうでないと仮定して、 b_{i-1} が a_1, \dots, a_n のどれと等しいかによって場合分けする。

- ある $k \leq i-2$ に対して $b_{i-1} = a_k$ と仮定する。 $a_k = \boxed{(95)} \boxed{(96)}$ なので $\boxed{(95)} \boxed{(96)} = b_{i-1}$ となり、与えられた数がすべて異なるという仮定に反する。
- $b_{i-1} = a_{i-1}$ と仮定する。すべての $k \leq \boxed{(97)} \boxed{(98)}$ に対して $a_k = b_k$ となるから、 $A < B$ であるためには $(a_i, \dots, a_n) < (b_i, \dots, b_n)$ でなければならない。ところが a_i, \dots, a_n は降順であるから補題

よりそのような b_i, \dots, b_n は存在しない。

- ある k があって $i \leq k \leq j-1$ かつ $b_{i-1} = a_k$ と仮定する。 a_i, \dots, a_n は降順であるから $a_k > a_j$ となる。すると $\boxed{(99)} \boxed{(100)} = a_j < a_k = b_{i-1}$ であるから $C < B$ 、したがって $A < C < B$ となり、 B が辞書順で A の次の順列という仮定に反する。
- ある k があって $\boxed{(101)} \boxed{(102)} \leq k \leq n$ かつ $b_{i-1} = a_k$ と仮定する。 a_j の選び方と、 a_i, \dots, a_n が降順であることから $a_{i-1} > a_k$ となる。すると $a_{i-1} > a_k = b_{i-1}$ となり、 $A < B$ という仮定に反する。

上の 4 つの場合はすべて仮定に反するので、 $b_{i-1} = a_j$ であることがわかる。

B の残りの部分 b_i, \dots, b_n は、 a_{i-1}, \dots, a_n から a_j を除いた数を並べ替えたものである。その中で辞書順で最も前に来るのは、補題より昇順に並んでいる場合である。 a_i, \dots, a_n は降順であり、 a_j の選び方から、 a_j の代わりに a_{i-1} を入れた $a_i, \dots, a_{j-1}, a_{i-1}, a_{j+1}, \dots, a_n$ も降順となり、それを逆にした $a_n, \dots, a_{j+1}, a_{i-1}, a_{j-1}, \dots, a_i$ が昇順となるので、これが b_i, \dots, b_n と一致する。

以上のことから $B = (a_1, \dots, a_{i-2}, a_j, a_n, \dots, a_{j+1}, a_{i-1}, a_{j-1}, \dots, a_i)$ となるが、これはアルゴリズムの出力に他ならない。

【 $\boxed{(89)} \boxed{(90)} \sim \boxed{(101)} \boxed{(102)}$ の選択肢】

- | | | | | |
|--------------|----------------|--------------|------------|---------------|
| (11) b_i | (12) b_{i-1} | (13) b_j | (14) b_k | (15) $b_{k'}$ |
| (16) c_i | (17) c_{i-1} | (18) c_j | (19) c_k | (20) $c_{k'}$ |
| (21) $i - 1$ | (22) i | (23) $i + 1$ | | |
| (24) $j - 1$ | (25) j | (26) $j + 1$ | | |