

情報1 2学期中間考査

福井県立勝山高等学校

2024年10月15日 3限目

注意事項

- 開始のチャイムが鳴るまで開かないこと.
- チャイムの前に問題用紙・解答用紙に記名して良い.
- 終了後, 問題冊子は持ち帰ること.

【マーク記入時の注意】

- 解答は, 解答用紙の問題番号に対応した解答欄にマークすること.
- (ウ) (エ) などの に数字を入れる場合, 以下の通りに記入せよ.

(例1) (ウ) (エ) に 34 と解答する場合.

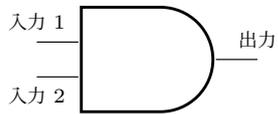
(ウ) に 3 を, (エ) に 4 を記入.

(例2) (ウ) (エ) に 1 と解答する場合.

(ウ) に 0 を, (エ) に 1 を記入.

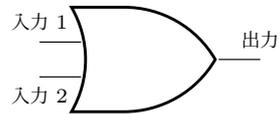
論理回路, フローチャート, python の例文については, 以下を参考にすること.

(A)AND 回路



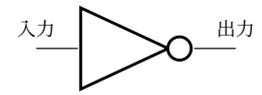
入力 1	入力 2	出力
0	0	0
0	1	0
1	0	0
1	1	1

(B)OR 回路



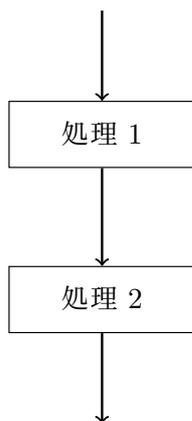
入力 1	入力 2	出力
0	0	0
0	1	1
1	0	1
1	1	1

(C)NOT 回路



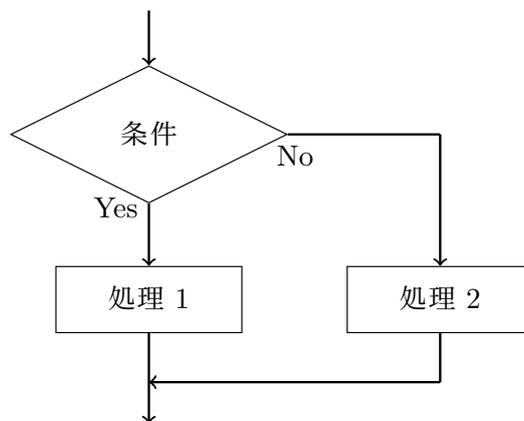
入力	出力
0	1
1	0

(A) 順次構造



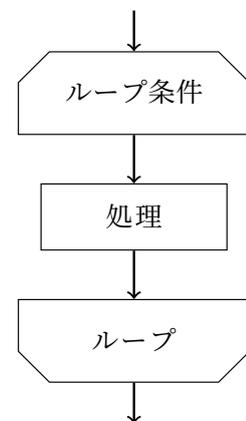
1つの処理の終了後に,
次の処理へと移る.

(B) 分岐構造



条件によって処理を選択し, 実行する.

(C) 反復構造



条件を満たす間,
同じ処理を繰り返し続ける.

python コード 基本文

1) 表示

```
1 print("Hello World")
2 print(321)
```

実行結果

```
1 Hello World
2 321
```

2) 繰り返し (for 文)

```
1 for i in range(3):
2     print(i, "Hello")
```

実行結果

```
1 0 Hello
2 1 Hello
3 2 Hello
```

3) 繰り返し (while 文)

```
1 i=0
2 while i<3:
3     print(i, "Hello")
4     i=i+1
```

実行結果

```
1 0 Hello
2 1 Hello
3 2 Hello
```

4) 条件分岐

```
1 a=int(input())
2 if a==1:
3     print("Hello")
4 elif a==2:
5     print("Bye")
6 else:
7     print("See You")
```

実行結果

実行後 2 を入力した場合

```
1 Bye
```

5) 演算

	演算子	記入例	結果
足し算	+	$a + b$	7
引き算	-	$a - b$	3
掛け算	*	$a * b$	10
割り算	/	a/b	2.5
商	//	$a//b$	2
余り	%	$a\%b$	1
累乗	**	$a ** b$	25

1 小問集合【30点】

(ア) マスメディアとは何か.

- ① 大手企業のメディアのこと.
- ② 新聞会社のこと.
- ③ 不特定多数の受信者に情報発信するメディアのこと.
- ④ 人々が基本的に備えているコミュニケーション手段の総称.

(イ) ユーザインタフェースの例として適切なものを選べ.

- ① 手
- ② CPU
- ③ マウス
- ④ 足

(ウ) 情報の可視化の例として適切なものを選べ.

- ① 文書
- ② ピクトグラム
- ③ 地図
- ④ 表

(エ) 情報の抽象化の例として適切なものを選べ.

- ① グラフ
- ② 文書
- ③ 地図
- ④ インフォグラフィックス

(オ) 情報の整理の5つの基準のうち、カテゴリ基準で並べられているものの例として適切なものを選べ.

- ① 辞書
- ② 地図
- ③ 図書館
- ④ 座席表

(カ) 公共施設のゴミ箱は、種類によって投入口の形が異なる。このようなものをなんというか。

- ① シグニファイア
- ② バリアフリー
- ③ ユーザエクスペリエンス
- ④ ユニバーサルデザイン

(キ) ユニバーサルデザインの7原則に当てはまらないものを選べ。

- ① 誰が見てもすぐに理解できる。
- ② 大きさや広さが十分である。
- ③ 直感的に違いがわかる。
- ④ 使用の自由度が低い。

(ク) 次のデータ量を計算せよ。ただし、1B=8bit, 1KB=1024B とする。

256 秒の音データで、標本化周波数は 64[Hz], 量子化ビット数は 32.

- ① 8KB
- ② 16KB
- ③ 64KB
- ④ 128KB

(ケ) 次のデータ量を計算せよ。ただし、1B=8bit, 1KB=1024B, 1MB=1024KB とする。

1280×960 のフルカラー動画 10 秒間、フレームレートは 30fps. また、フルカラー画像は 1 画素あたり 24bit とする。

- ① 100000 MB
- ② 10000 MB
- ③ 1000 MB
- ④ 100 MB

(コ) CD の標本化周波数は、44.1kHz である。このことから予想される人の最大可聴周波数はどれか。

- ① 22kHz
- ② 40kHz
- ③ 44kHz
- ④ 88kHz

(サ) クラウドストレージについて述べた文として正しいものを選べ。

- ① 補助記憶装置であり、データセンタに存在することが多い。
- ② 物理的な記憶装置はないため、データが失われることがない。
- ③ ネットワーク接続していなくても利用可能である。
- ④ 操作している端末に組み込まれている補助記憶装置である。

(シ) 以下のコードの実行結果として適切なものを選べ。

```
1 for i in range(5):
2     for j in range(10-2*i):
3         print("-",end="")
4     for j in range(3*i+1):
5         print("*",end="")
6     print("")
```

注) end="" と書くことで、改行をキャンセルすることができる。

①

```
1 -----
2 -----**
3 -----*****
4 ----*****
5 --*****
```

②

```
1 -----*
2 -----****
3 -----*****
4 ----*****
5 --*****
```

③

```
1 -----*
2 -----****
3 -----*****
4 -----*****
5 -----*****
```

④

```
1 -----*
2 -----****
3 -----*****
4 ----*****
5 --*****
6 *****
```

(ス) 以下のコードの実行結果として適切なものを選び。

```
1 for i in range(5):
2     for j in range(6):
3         if (i*j)%2==0:
4             print("*",end="")
5         else:
6             print("--",end="")
7     print("")
```

①

```
1 *****
2 *--*--*--
3 *****
4 *--*--*--
5 *****
```

②

```
1 *--*--*--
2 --*--*--*
3 *--*--*--
4 --*--*--*
5 *--*--*--
```

③

```
1 *****
2 *--*--
3 *****
4 *--*--
5 *****
```

④

```
1 -----
2 --*--*--*
3 -----
4 --*--*--*
5 -----
```

(セ) $10101_{(2)} + 11100_{(2)}$ の結果として適切なものを選べ.

- ① $110010_{(2)}$
- ② $101001_{(2)}$
- ③ $110001_{(2)}$
- ④ $100001_{(2)}$

(ソ) $101_{(2)} \times 11_{(2)}$ の結果として適切なものを選べ.

- ① $1111_{(2)}$
- ② $1011_{(2)}$
- ③ $11011_{(2)}$
- ④ $11001_{(2)}$

問題は続きます.

2 【データ圧縮：40点】

[1] ランレングス圧縮

AさんとBさんが、授業で学んだデジタルデータについて会話している。
以下を読んで、右ページからの問いに答えよ。

Aさん「授業では、文字・音声・画像・動画のデジタル表現について学んだね。」

Bさん「そうだったけ。」

Aさん「話聞いてたの。あなた、テストやばいよ。」

Bさん「僕は天才だから大丈夫さ。」

Aさん「へえ。君のことなんて興味ないね。」

Bさん「ちなみに、ランレングス圧縮ってなんだっけ。」

Aさん「ほら、理解できてないじゃん。私は教えない。君みたいな人はテストで赤点取ればいいんだよ。」

Cさん「親切な私が教えてあげるね。たぶんAさんも分かってないんだよ。」

ランレングス圧縮の手法

ランレングス圧縮は、(ア)の手法のひとつで、繰り返し現れる構造を省略する圧縮である。



上の画像であれば、色は2色なので色を表すために1bit必要である。そのため、元データは(イ)(bit)である。

一方圧縮後は、同色は最大で16個並ぶ可能性があるので、数を表すために4bit必要。そのため、圧縮後は、

$$1 + 4 + 1 + 4 + 1 + 4 = 15(\text{bit})$$

である。

Aさん「このくらい簡単だよな。」

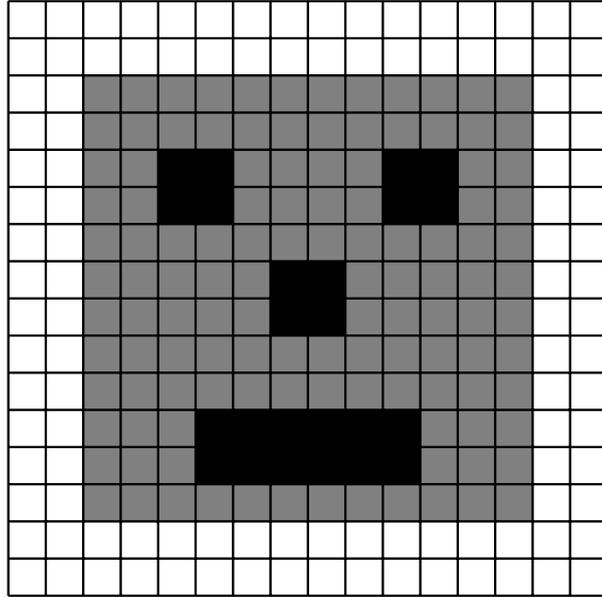
Bさん「... Cさん、ありがとう。」

Cさん「何か困ったらいつでも頼ってね。」

Bさん「じゃあ、もう少し教えてよ。」

Cさん「ええよ。」

ランレングス圧縮手法 2



	16			
	16			
	2	14		2
	2	14		2

⋮

各行ごとに、前ページで行った圧縮を行う。この画像では色が3色あるため、色を表す部分で (ウ) (bit) 必要。そのほかは、前述の通り。

Bさん「ちなみに、圧縮率ってどうやって計算するんだっけ。」

Cさん「 $(\text{圧縮率}) = \frac{(\text{圧縮後のデータ量})}{(\text{元のデータ量})}$ で計算できるよ。」

Bさん「ということは、圧縮率 100% って、 (エ)」

Cさん「そうなんだよ。いい感じで理解できてきたね。」

Bさん「Cさんのおかげだよ。」

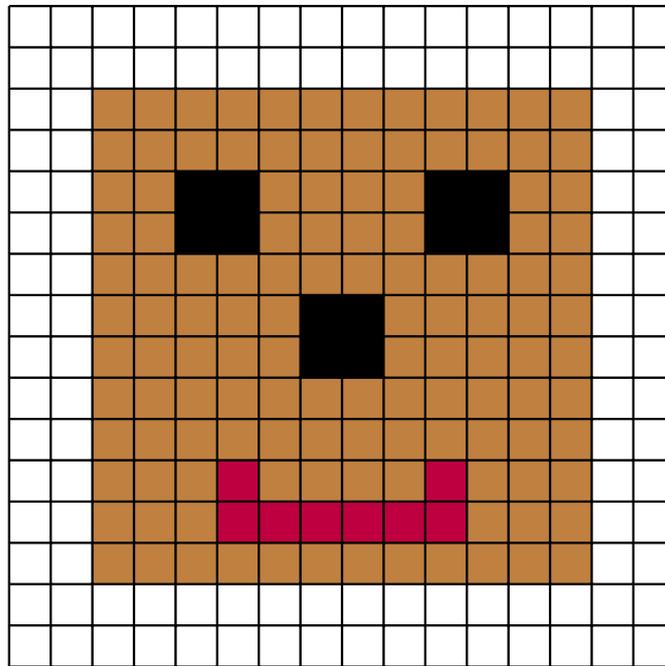
【問題】

文章中の(ア)~(エ)に当てはまるものとして、最も適切なものを以下の選択肢から選べ。

選択肢

- ①. 可逆圧縮 ②. 非可逆圧縮 ③. 音声の圧縮
- ④. 万能な圧縮 ⑤. よく圧縮できている. ⑥. 圧縮できていない.
- ⑦. 存在しない. ⑧. 1 ⑨. 2
- ⑩. a. 4 b. 8 c. 16
- ⑪. d. 32

(オ) 以下の画像をランレングス圧縮した際の圧縮率として適切な数値を選べ。



- ①. 64%
- ②. 68%
- ③. 72%
- ④. 76%

大問2 は続きます.

[2] ハフマン符号

ハフマン符号は、可変長符号である。符号化方法は、以下の通りである。

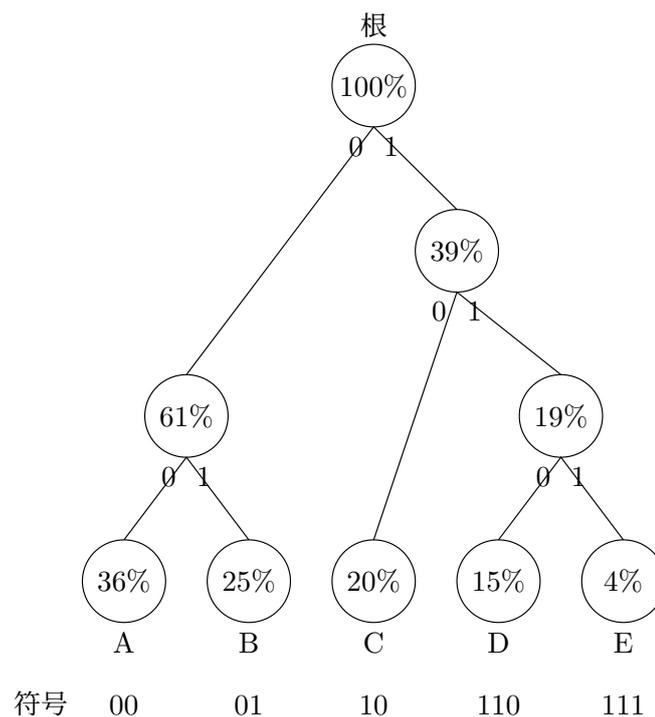
ハフマン符号

【手順】

- (1) 出現頻度の多い順に文字を並べる。
- (2) 出現頻度の少ない 2 つを選んで、枝で結ぶ。枝の根に、2 つの出現頻度を足した値を書き添える。
- (3) 全ての枝が繋がるまで (2) の操作を続ける。
- (4) 根からスタートして、左に 0 を、右に 1 を割り当て、根から目的の文字までの 0 と 1 を書き並べたものが、ハフマン符号。

【例】

文字	A	B	C	D	E
出現頻度	36%	25%	20%	15%	4%



固定長符号の場合、5 文字を表すためには (カ) (bit) 必要であり、元データが 100 文字の場合、 (カ) (bit) × 100 (文字) = (bit) のデータ量となる。

一方、ハフマン符号で符号化した場合、2(bit) が 81 文字、3(bit) が 19 文字であるため、圧縮率は (キ) % である。

【問題】

(カ) 当てはまる数値を答えよ.

(キ) 当てはまる数値を選べ.

- ① 70
- ② 73
- ③ 76
- ④ 79

- ある文書を読み取ったところ, 各文字の出現率は以下の通りであった. ハフマン符号化した場合について, 各問いに答えよ.

文字	A	B	C	D	E	F	G
出現頻度	35%	3%	6%	6%	25%	5%	20%

(ク) 文字「A」の符号として適切なものを選べ.

- ① 00
- ② 10
- ③ 01
- ④ 101

(ケ) 文字「F」の符号として適切なものを選べ.

- ① 0100
- ② 1100
- ③ 1110
- ④ 0110

(コ) 固定長符号の場合と比較した際の圧縮率を求めよ.

- ① 60%
- ② 70%
- ③ 80%
- ④ 90%

3 【プログラミング：30点】

以下の会話文を読んで、各問いに答えよ。

A さん 授業で「探索」と「ソート」を学んだよ。

B さん プログラミングの授業だね。二分岐探索について教えてよ。

A さん ええで。手順書を教えるね。

手順書

昇順に並んだ配列 a と探している数字が与えられたとき、以下の手順で探索する。

- (1) 探索範囲の左端の配列番号を $left$ 、右端の配列番号を $right$ で初期化する。
- (2) $left$ の値が $right$ の値以下の場合、以下を繰り返す。
 - i. $(left+right)\div 2$ の商の値を mid とする。(中央の値を得る)
 - ii. mid 番目の配列が調べている数であれば、その配列番号を出力してループ脱出。
 - iii. mid 番目の配列 $<$ 調べている数であれば、 $left$ を $mid+1$ にする。
 - iv. mid 番目の配列 $>$ 調べている数であれば、 $right$ を $mid-1$ にする。
- (3) $left > right$ の場合、Not Found と出力する。

B さん なるほど。じゃあ、これをコード化するとこんな感じだね。

```
1 data=[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2 search_num=int(input("search number=,))
3 left=0
4 right=len(data)-1
5 while left<=right:
6     mid=(left+right)//2
7     if data[mid]==search_num:
8         print("No.",mid)
9         break
10    elif data[mid]<search_num:
11        left = mid + 1
12    else:
13        right = mid - 1
14 if left>right:
15    print("Not found")
```

A さん すごい。私が構造理解できているか質問してくれないかい。

B さん ええで。1 問目: 5 行目を別の表現で表してみて。

A さん

B さん 正解。では、2 問目: 5 と入力した場合、表示される結果を答えてみて。

A さん

B さん 正解。いい感じに理解できているね。じゃあ、3 問目: 100 と入力した際、「right」の値は最終的に何になるでしょう。

A さん

B さん 正解。最終問題だよ。4 問目: 1 と入力した際、「mid」の数値はどのように変化していくでしょう。

A さん

B さん ちなみに、この探索手法は昇順に並んでいないとできないのかな。

A さん 降順でもできるよ。その場合は、コードを少し変更すればいいよ。

B さん そうなんだね. 昇順, 降順に並べるプログラムについても学んだね.

A さん ソートアルゴリズムだね. 授業では「バブルソート」「選択ソート」について学んだね.

B さん バブルソートについて復習しよう.

手順書

配列 L を以下の手順で並べ替えていく.

(1) 以下の操作を $i = 0$ から (配列長 -1) 回繰り返す.

i. 以下の操作を $j = 0$ から ((配列長) $- i - 1$) 回繰り返す.

A. j 番目の数 $> j + 1$ 番目の数であれば数の入れ替え

B. そうでなければ何もしない.

```
1 L=[2,3,1,0,7,9,6,5,10]
2 for i in range(len(L)-1):
3     for j in range(len(L)-i-1):
4         if L[j]>L[j+1]:
5             tmp=L[j+1]
6             L[j+1]=L[j]
7             L[j]=tmp
8 print(L)
```

A さん プログラム中に tmp とあるのはなぜ.

B さん tmp は temporary の略 3 文字で, 入れ替えの際の一時的な数値保管場所だと思えばいいよ.

A さん ちょっと詳しく教えてよ.

B さん 下のようなコードを実行してみよう.

```
1 L=[2,3]
2 L[0]=L[1]
3 L[1]=L[0]
4 print(L)
```

すると, と表示されるんだ.

A さん だから必要なんだね. 完全に理解したよ.

B さん じゃあ, 2 行目と 3 行目の間に「print("i=",i," L=", L)」を入れた場合, どのような出力になるかな.

A さん

B さん さすがだね. 正解だよ.

A さん これで私も凄腕のエンジニアになれるね.

B さん それは無理だね. 君にはまだ早いよ.

次のページからの問題に答えよ.

【問題】

(ア) 当てはまるものとして最も適切なものを答えよ.

- ① 「while left<right:」
- ② 「for i in range(len(L)):」
- ③ 「while right>left-1:」
- ④ 「while left+1<right:」

(イ) 当てはまるものとして最も適切なものを答えよ.

- ① 「4」
- ② 「5」
- ③ 「6」
- ④ 「Not found」

(ウ) 当てはまるものとして最も適切なものを答えよ.

- ① 0
- ② -1
- ③ 1
- ④ 9
- ⑤ 10
- ⑥ 11

(エ) 当てはまるものとして最も適切なものを答えよ.

- ① 「 $5 \rightarrow 2 \rightarrow 1$ 」
- ② 「 $5 \rightarrow 2 \rightarrow 0 \rightarrow 1$ 」
- ③ 「 $6 \rightarrow 2 \rightarrow 0 \rightarrow 1$ 」
- ④ 「 $6 \rightarrow 2 \rightarrow 1$ 」

(オ) 下線部について、降順に並べた配列に対して二分探索が適切に動くコードを1つ選べ。
(1行目の配列は全て変更。その他変更箇所は表記の通り。)

① 変更なし。

```
1 data=[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
2 search_num=int(input("search number=,))
3 left=0
4 right=len(data)-1
5 while left<=right:
6     mid=(left+right)//2
7     if data[mid]==search_num:
8         print("No.",mid)
9         break
10    elif data[mid]<search_num:
11        left = mid + 1
12    else:
13        right = mid - 1
14 if left>right:
15    print("Not found")
```

② 5行目「while left<=right:」 → 「while left>=right:」

```
1 data=[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
2 search_num=int(input("search number=,))
3 left=0
4 right=len(data)-1
5 while left>=right:
6     mid=(left+right)//2
7     if data[mid]==search_num:
8         print("No.",mid)
9         break
10    elif data[mid]<search_num:
11        left = mid + 1
12    else:
13        right = mid - 1
14 if left>right:
15    print("Not found")
```

③ 5行目「while left<=right:」 → 「while left>=right:」

10行目「elif data[mid] <search_num:」 → 「elif data[mid] >search_num:」

```
1 data=[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
2 search_num=int(input("search number=",))
3 left=0
4 right=len(data)-1
5 while left>=right:
6     mid=(left+right)//2
7     if data[mid]==search_num:
8         print("No.",mid)
9         break
10    elif data[mid]>search_num:
11        left = mid + 1
12    else:
13        right = mid - 1
14 if left>right:
15    print("Not found")
```

④ 10行目「elif data[mid] <search_num:」 → 「elif data[mid] >search_num:」

```
1 data=[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
2 search_num=int(input("search number=",))
3 left=0
4 right=len(data)-1
5 while left<=right:
6     mid=(left+right)//2
7     if data[mid]==search_num:
8         print("No.",mid)
9         break
10    elif data[mid]>search_num:
11        left = mid + 1
12    else:
13        right = mid - 1
14 if left>right:
15    print("Not found")
```

(カ) 当てはまるものとして最も適切なものを選べ.

①

1 [2, 2]

②

1 [3, 3]

③

1 [2, 3]

④

1 [3, 2]

(キ) B さんはの指示を反映させたコードは以下の通りである. 出力結果として適切なものを選び.

```
1 L=[2,3,1,0,7,9,6,5,10]
2 for i in range(len(L)-1):
3     print("i=",i," L=", L)
4     for j in range(len(L)-i-1):
5         if L[j]>L[j+1]:
6             tmp=L[j+1]
7             L[j+1]=L[j]
8             L[j]=tmp
9 print(L)
```

①

```
1 i= 0 , L= [2, 3, 1, 0, 7, 9, 6, 5, 10]
2 i= 1 , L= [2, 1, 0, 3, 7, 6, 5, 9, 10]
3 i= 2 , L= [1, 0, 2, 3, 6, 5, 7, 9, 10]
4 i= 3 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
5 i= 4 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
6 i= 5 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
7 i= 6 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
8 i= 7 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
9 [0, 1, 2, 3, 5, 6, 7, 9, 10]
```

②

```
1 i= 0 , L= [2, 1, 0, 3, 7, 6, 5, 9, 10]
2 i= 1 , L= [1, 0, 2, 3, 6, 5, 7, 9, 10]
3 i= 2 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
4 i= 3 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
5 i= 4 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
6 i= 5 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
7 i= 6 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
8 i= 7 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
9 [0, 1, 2, 3, 5, 6, 7, 9, 10]
```

③

```
1 i= 1 , L= [2, 3, 1, 0, 7, 9, 6, 5, 10]
2 i= 2 , L= [2, 1, 0, 3, 7, 6, 5, 9, 10]
3 i= 3 , L= [1, 0, 2, 3, 6, 5, 7, 9, 10]
4 i= 4 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
5 i= 5 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
6 i= 6 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
7 i= 7 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
8 i= 8 , L= [0, 1, 2, 3, 5, 6, 7, 9, 10]
9 [0, 1, 2, 3, 5, 6, 7, 9, 10]
```

以上で問題は終了です.

