

次の問8は必須問題です。必ず解答してください。

問8 次のプログラムの説明及びプログラムを読んで、設問1, 2に答えよ。

二つの文字列の差異を測る指標に編集距離がある。編集距離の概念は、文書比較や検索キーワードの候補の提示などに用いられている。編集距離とは、1文字の追加操作又は削除操作を繰り返し適用し、ある文字列を別の文字列に変換するのに必要な最小の操作回数である。例えば文字列"abcabba"を文字列"cbabac"に変換する場合、図1に示す操作1～5によって変換が完了する。図1は、最小の操作回数で変換する一例を示しており、編集距離は5となる。

abcabba	→	bcabba (1文字目のaを削除)	操作1
	→	cabba (1文字目のbを削除)	操作2
	→	cbabba (1文字目のcの後ろにbを追加)	操作3
	→	cbaba (5文字目のbを削除)	操作4
	→	cbabac (5文字目のaの後ろにcを追加)	操作5

図1 文字列"abcabba"を文字列"cbabac"に変換する場合の例

関数 CalcEditDistance は、二つの文字列間の編集距離を返す関数である。

- (1) 変換元の文字列を Str1[], 変換先の文字列を Str2[]とする。また、配列の添字は0から始まり、文字列Str1[]のi番目の文字はStr1[i - 1]と表記する。したがって、Str1[] = "abcabba"の場合、1番目の文字 = Str1[0] = "a", 2番目の文字 = Str1[1] = "b"となる。Str2[]についても同様である。
- (2) 関数 CalcEditDistance は、エディットグラフと呼ばれるグラフの最短距離取得問題の考え方に基づいて、編集距離を求めている。編集距離の求め方は次のとおりである。
 - ① 次の手順で、xy平面上にエディットグラフを作成する。ここで、Str1Len はStr1[]の文字数、Str2LenはStr2[]の文字数である。
 - (a) $0 \leq X \leq \text{Str1Len}$ を満たす全ての整数Xに対して、点(X, 0)から点(X, Str2Len)に線分を引く。

- (b) $0 \leq Y \leq \text{Str2Len}$ を満たす全ての整数 Y に対して、点 $(0, Y)$ から点 $(\text{Str1Len}, Y)$ に線分を引く。
- (c) $0 \leq X < \text{Str1Len}$, $0 \leq Y < \text{Str2Len}$ を満たす全ての整数 X, Y の組に対して、 $\text{Str1}[X]$ と $\text{Str2}[Y]$ が同一の文字の場合、点 (X, Y) から点 $(X + 1, Y + 1)$ に線分を引く。
- ② エディットグラフを構成する線分をたどって、点 $(0, 0)$ から点 $(\text{Str1Len}, \text{Str2Len})$ へ移動する経路を考える。点 (X, Y) から点 $(X + 1, Y)$ 又は点 $(X, Y + 1)$ への移動距離を 1、点 (X, Y) から点 $(X + 1, Y + 1)$ への移動距離を 0 としたときの、点 $(0, 0)$ から点 $(\text{Str1Len}, \text{Str2Len})$ までの最短移動距離が編集距離となる。
- (3) $\text{Str1}[] = "abcabba"$, $\text{Str2}[] = "cbabac"$ の場合のエディットグラフを図 2 の左に示す。この場合に、最短移動距離となる経路の一つを図 2 の右に → で示す。

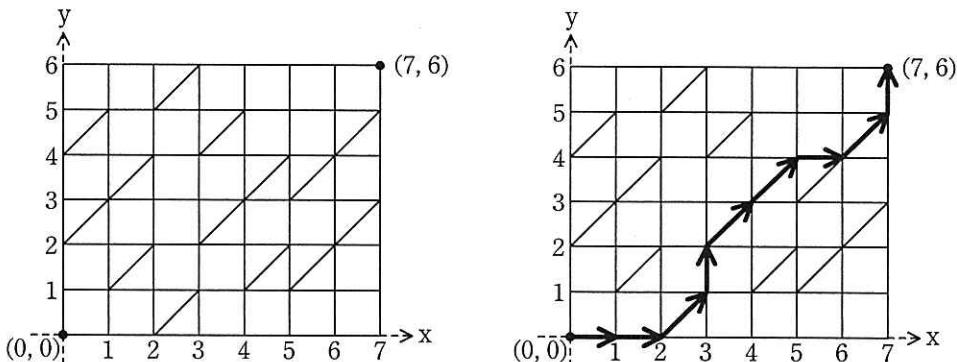


図 2 $\text{Str1}[] = "abcabba"$, $\text{Str2}[] = "cbabac"$ の場合の
エディットグラフ（左）と最短移動距離となる経路の例（右）

- (4) 関数 `CalcEditDistance` では、点 $(0, 0)$ から点 (X, Y) への最短移動距離を $D[X, Y]$ に求めている。 $D[X, Y]$ は、既に算出されている $D[X - 1, Y - 1]$, $D[X, Y - 1]$, $D[X - 1, Y]$ を用いて求めることができる。これによって編集距離を算出している。

[関数 CalcEditDistance の引数と返却値]

関数 CalcEditDistance の引数の仕様は、次のとおりである。各配列の添字は、0 から始まる。

引数名／返却値	データ型	意味
Str1[]	文字型	変換元の文字列が格納されている 1 次元配列
Str1Len	整数型	変換元の文字列の長さ (1 以上)
Str2[]	文字型	変換先の文字列が格納されている 1 次元配列
Str2Len	整数型	変換先の文字列の長さ (1 以上)
返却値	整数型	変換元と変換先の文字列間の編集距離

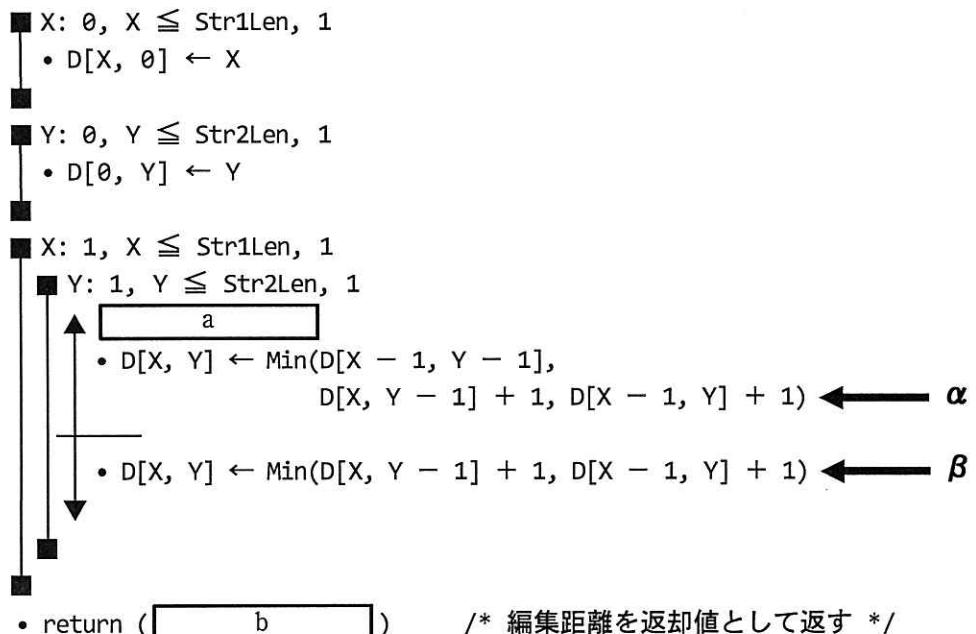
関数 CalcEditDistance では、次の関数 Min を使用している。

[関数 Min の仕様]

引数として与えられた二つ以上の整数値の中で最も小さい値を返却値とする。

[プログラム]

- 整数型: CalcEditDistance(文字型: Str1[], 整数型: Str1Len,
 文字型: Str2[], 整数型: Str2Len)
- 整数型: D[Str1Len + 1, Str2Len + 1], X, Y



設問1 プログラム中の に入る正しい答えを、解答群の中から選べ。

aに関する解答群

- ア $\text{Str1}[X - 1] = \text{Str2}[Y - 1]$
- イ $\text{Str1}[X - 1] \neq \text{Str2}[Y - 1]$
- ウ $\text{Str1}[X] = \text{Str2}[Y]$
- エ $\text{Str1}[X] \neq \text{Str2}[Y]$
- オ $\text{Str1}[X - 1] = \text{Str1}[X] \text{ and } \text{Str2}[Y - 1] = \text{Str2}[Y]$
- カ $\text{Str1}[X - 1] \neq \text{Str1}[X] \text{ and } \text{Str2}[Y - 1] \neq \text{Str2}[Y]$

bに関する解答群

- ア $D[0, \text{Str2Len}]$
- イ $D[\text{Str1Len}, 0]$
- ウ $D[\text{Str1Len}, \text{Str2Len}]$
- エ $D[\text{Str1Len}, \text{Str2Len}] + 1$
- オ $D[\text{Str1Len} - 1, \text{Str2Len} - 1]$
- カ $D[\text{Str1Len} - 1, \text{Str2Len} - 1] + 1$

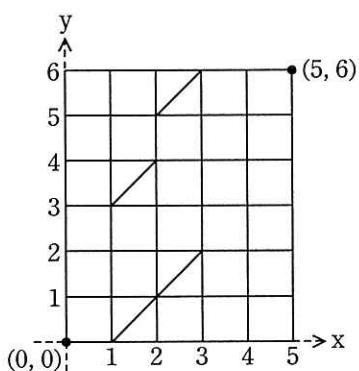
設問2 次の記述中の に入る正しい答えを、解答群の中から選べ。

$\text{Str1[]} = \text{"peace"}, \text{Str2[]} = \text{"people"}$ の場合のエディットグラフは、
となる。

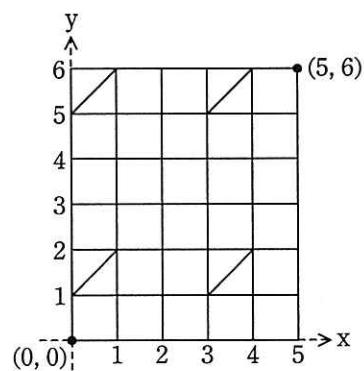
`CalcEditDistance("peace", 5, "people", 6)`を実行した場合、関数
`CalcEditDistance`が終了するまでに行 α は 回実行され、行 β は
 回実行される。また、返却値は となる。ここで、関数
`CalcEditDistance`中の , には正しい答えが入っているものとする。

cに関する解答群

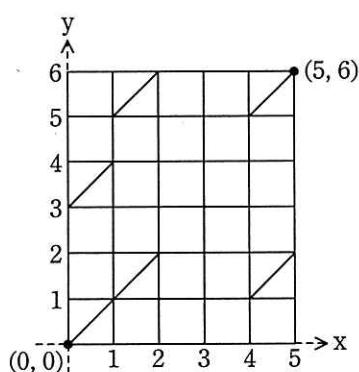
ア



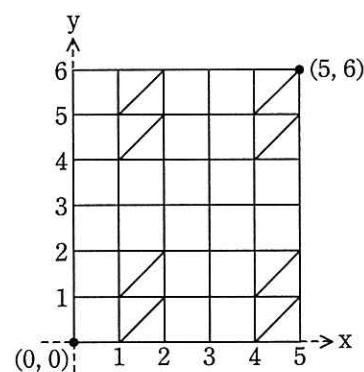
イ



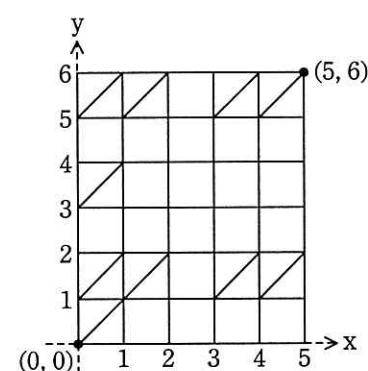
ウ



エ



オ



dに関する解答群

ア 2

イ 4

ウ 6

エ 8

オ 10

カ 12

キ 14

ク 16

e に関する解答群

ア 14	イ 16	ウ 18	エ 20
オ 22	カ 24	キ 26	ク 28

f に関する解答群

ア 4	イ 5	ウ 6	エ 7
オ 8	カ 9	キ 10	ク 11