問20　次のプログラム中の　　　　　　に入れる正しい答えを、解答群の中から選べ　T

　　 ここで、配列の添え字は0から始まる。

　　手続きconvert(整数型：num, 文字型の配列：str)は、金額を表す時のように、整数を３桁区切り形式の文字

列に変換する。

1. 次のルールに基づいて変換を行う。
   1. 整数値が負の場合、先頭にマイナス符号をつける。
   2. 整数値の下から3桁ごとにコンマを挿入する。

変換例を表に示す。

　●表　変換例

|  |  |
| --- | --- |
| 整数 | 3桁区切り形式の文字列 |
| 1234567 | 1,234,567 |
| －57482 | －57,482 |
| 63 | 63 |
| －999999 | －999,999 |

1. 手続きconvert(整数型：num, 文字型の配列：str)の仕様は、次のとおりである。

引数：整数num、変換後の文字列str

返却値：なし

機能：整数numを3桁区切り形式の文字列に変換してstrに先頭から格納する。

　　　strの末尾に、システム定数EOSを格納する。

　　　strには変換後の文字列を格納するのに十分な領域が確保されている。

　　　整数numは9桁以下とする。

　[プログラム]

〇convert(整数型：num, 文字型の配列：str)

　　論理型：minus

　　整数型：i , j

　　文字型の配列：table

　　文字型：tmp

　　minus ← false

　　i ← 0

　　j ← 0

　　table ← “01234567”

解答群

ア　i mod 3

イ　( i + 1 ) mod 3

ウ　( i + 2 ) mod 3

エ　( i － 1 ) mod 3

　　/\* 整数値が負かどうかをチェックする \*/

　　if( num < 0 )

　　　　minus ← true

num ← －num

endif

/\* 数値の下位から順に文字に変換する \*/

　　do

/\* 1桁の文字を格納する \*/

　　　　str[j] ← table[num mod 10]

j ← j + 1

/\* 次のけたを取り出す準備をする \*/

　　　　num ← num ÷ 10

/\* 格納した文字の数をカウントアップする \*/

　　　　i ← i + 1

/\* 3桁ごとに区切りのカンマを挿入する \*/

　　　　if( 　　　　　 = 0 and num ≠ 0 )

str[j] ← “ , ”

j ← j + 1

endif

while( num ≠ 0 )

　　/\* 整数値が負ならマイナス符号を付ける \*/

　　if( minus = true )

str[j] ← “－”

j ← j + 1

endif

str[j] ← EOS

/\* strの要素の順序を逆にする \*/

　　i ← 0

j ← j － 1

　　while( i < j )

tmp ← str[i]

str[i] ← str[j]

str[j] ← tmp

i ← i + 1

j ← j － 1

endwhile