

問1 次のプログラムの説明及びプログラムを読んで、 に入れる適切な字句を、解答群の中から選べ。

<プログラムの説明>

コマンドラインで指定された、1,000 以下の一つの自然数を素因数分解し、自然数の素因数をすべて表示するプログラムである。

1. コマンドラインから 2 以上、1,000 以下の整数を指定して実行する。このとき、必ず一つの整数を指定するものとする。ここで、1,000 より大きい数が指定された場合には、メッセージを表示してプログラムを終了する。また、コマンドラインから指定された値が整数ではない場合及び 1 以下の場合は、考慮しないものとする。
2. 素因数を求める手順は、以下のとおりである。
 - ① あらかじめ素数を小さい順に用意しておく、最初に、一番小さい素数を除数とする。
 - ② コマンドラインから指定された値を N とする。除数が N の平方根の値以下である間は、次の③から④を繰り返す。
 - ③ N が除数で割り切れるか否かを判定する。
 - ④ 割り切ることができたら、除数を素因数として表示し、N を除数で割った商を、新たな N とする。

なお、割り切れなかった場合は、次に大きい素数を除数とする。

- ⑤ 繰り返し処理の終了後に、N の最後の値も素因数なので表示する。

《実行例》

```
>java Q5 18
18 : 2 3 3
>java Q5 388
388 : 2 2 97
>java Q5 1001
1,000 以下の整数を入力してください。
```

<プログラム>

```
1 public class Q5 {
2     public static void main(String[] args) {
3         // 割るための素数を用意
4         int sosuu[] =
5             { 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37 };
6
7         // 入力された整数を変換&チェック
8         int num;
9         num =  (28);
10
11        if (  (29) ) {
12            System.out.println( "1,000 以下の整数を入力してください。" );
13            return;
14        }
15
16        System.out.print( num + " :");
17
18        // 素因数分解
19        int i = 0;
20        int n = num;
21        while ( sosuu[i] <= Math.sqrt( n ) ) {
22            if (  (30) ) {
23                // 素数で割りきれる場合
24                System.out.print( " " +  (31) );
25                n = n / sosuu[i];
26            } else {
27                // 素数で割りきれない場合
28                 (32);
29            }
30        }
31
32        // 最後に残った数値も素因数なので表示
33        System.out.print( " " + n );
34    }
35 }
```

(28) の解答群

- ア args
イ args[0]
ウ Integer.parseInt(args)
エ Integer.parseInt(args[0])

(29) の解答群

- ア num > 1000
イ num < 1000
ウ num >= 1000
エ num <= 1000

(30) の解答群

- ア n / sosuu[i] == 0
イ n / sosuu[i] != 0
ウ n % sosuu[i] == 0
エ n % sosuu[i] != 0

(31) の解答群

- ア n
イ num
ウ sosuu[i]
エ sosuu[n]

(32) の解答群

- ア n++
イ num++
ウ i++
エ sosuu++

(28)	(29)	(30)	(31)	(32)

問 2 次のプログラムの説明及びプログラムを読んで、 に入れる適切な字句を、解答群の中から選べ。

<プログラムの説明>

このプログラムは、ハンバーガーショップで提供されるメニューのアレルギ－情報から、指定されたアレルギ－食品を含むメニューを抽出し、表示するプログラムである。なお、アレルギ－食品の指定は、コマンドラインから行う。

1. アレルギ－食品は「乳製品、小麦、そば、卵、落花生、えび、かに」の七つとし、それぞれに英字が一つ一つアレルギ－コードとして、次の表のように割り当てられている。

アレルギ－コード	アレルギ－食品
M	乳製品
W	小麦
B	そば
E	卵
P	落花生
S	えび
C	かに

2. アレルギ－情報は、アレルギ－コードを並べた一つの文字列で表される。

（例） 乳製品、小麦、卵を含む : “MWE”
えびを含む : “S”

3. メニューは複数あり、一つのメニューはその中に含まれるアレルギ－食品のアレルギ－情報とメニュー名で表される。
4. メニューは、複数のアレルギ－食品を含みうる。ただし、アレルギ－食品を含まないメニューもある。
5. コマンドラインから指定されたアレルギ－情報に含まれるアレルギ－食品を一つでも含むメニューは、表示の対象となる。
6. このプログラムでは、Q6 クラスと Menu クラスを定義する。各クラスの役割は、次のとおりである。

Q6 クラス : 実行用クラス
Menu クラス : メニューに関する情報を保持するクラス

7. Q6 クラス、Menu クラスにおけるメンバ変数とメソッドは、次のとおりである。

Q6	Menu
	String allergen (アレルギ－情報) String menuName (メニュー名)
main メソッド	contains メソッド (指定されたアレルギ－食品を含むかどうかを返す。)

《実行例》

>java Q6 WS
ハンバーガー
エビバーガー
ライスバーガー

8. プログラムで使用する API の説明を、以下に示す。

java.lang パッケージ
String クラス
public final class String String クラスは文字列を表す。String クラスには、文字列に関するさまざまな処理を行うメソッドがある。
boolean contains メソッド
boolean contains(CharSequence s) s には String を与えることができ、この文字列が指定された文字列 s を含む場合に true を返す。 戻り値：この文字列が s を含む場合は true。そうでない場合は false 例：String str = "ABC"; とすると、str.contains("B")は、true を返す。
char[] toCharArray メソッド
public char[] toCharArray() この文字列を新しい文字配列に変換する。 戻り値：新しく割り当てられた文字配列。長さはこの文字列の長さと同じで、内容はこの文字列によって表される文字シーケンスが格納されるように初期化されている。 例：String s = "ABC"; char c[] = s.toCharArray(); とすると、配列 c は、c[0] == 'A', c[1] == 'B', c[2] == 'C'となる。
String valueOf メソッド
String valueOf(char c) char 引数の文字列表現を返す。 戻り値：一つの文字として引数 c が格納された長さ 1 の文字列

問 2 （つづき）

<プログラム>

```
1 class Q6 {
2     public static void main(String[] args) {
3         String allergy_info[][] = {
4             { "MWEP", "ハンバーガー" },
5             { "MWEPS", "エビバーガー" },
6             { "W", "ライスバーガー" },
7             { "C", "クラムチャウダー" },
8             { "", "ウーロン茶" } };
9
10        Menu[] menues = new Menu[allergy_info.length];
11        for (int i = 0; i < allergy_info.length; i++) {
12            menues[i] = new Menu();
13            menues[i].allergen = allergy_info[i][0];
14            menues[i].menuName = (33);
15        }
16
17        for (Menu menuItem : (34) ) {
18            if (menuItem.contains( (35) )) {
19                System.out.println( menuItem.menuName );
20            }
21        }
22    }
23 }
24
25 class Menu {
26     String allergen;
27     String menuName;
28     boolean contains(String allergencode) {
29         boolean result = (36);
30         for (char code : allergencode.toCharArray()) {
31             if (allergen.contains( String.valueOf( code ))) {
32                 result = true;
33             }
34         }
35         return (37);
36     }
37 }
```

(33) の解答群

- ア allergy_info[0]
- イ allergy_info[1]
- ウ allergy_info[i][0]
- エ allergy_info[i][1]

(34) の解答群

- ア menuItem
- イ menues
- ウ allergy_info
- エ allergy_info[0]

(35) の解答群

- ア menuItem
- イ menues
- ウ args
- エ args[0]

(36) の解答群

- ア true
- イ false
- ウ null
- エ allergencode

(37) の解答群

- ア true
- イ false
- ウ result
- エ allergencode

(33)	(34)	(35)	(36)	(37)