

問題 3.2.1

答えは以下ようになります。分からない人は、**3.2.2項**に戻って確認しましょう。

ステップ数	0	1	2	3	4	5	6
A の値	372	372	104	104	14	14	0
B の値	506	134	134	30	30	2	2

問題 3.2.2

整数 A_1, A_2, \dots, A_N の最大公約数は、以下のようにして計算できます。（→**3.2.5項**）

- まず、 A_1 と A_2 の最大公約数を計算する。
- 次に、前の計算結果と A_3 の最大公約数を計算する。
- ：
- 次に、前の計算結果と A_N の最大公約数を計算する。

これを実装すると、以下ようになります。なお、関数 $\text{GCD}(A, B)$ は A と B の最大公約数を計算する関数です。また、変数 r は前の計算結果を表します。

```
#include <iostream>
#include <algorithm>
using namespace std;

long long GCD(long long A, long long B) {
    while (A >= 1 && B >= 1) {
        if (A < B) B = B % A; // A < B の場合、大きい方 B を書き換える
        else A = A % B; // A >= B の場合、大きい方 A を書き換える
    }
    if (A >= 1) return A;
    return B;
}

long long N;
long long A[100009];

int main() {
```

```

// 入力
cin >> N;
for (int i = 1; i <= N; i++) cin >> A[i];

// 答えを求める
long long R = GCD(A[1], A[2]);
for (int i = 3; i <= N; i++) {
    R = GCD(R, A[i]);
}

// 出力
cout << R << endl;
return 0;
}

```

※ Python などのソースコードは GitHub の codes フォルダをご覧ください。

問題 3.2.3

整数 A_1, A_2, \dots, A_N の最小公倍数は、以下のようにして計算できます。

- まず、 A_1 と A_2 の最小公倍数を計算する。
- 次に、前の計算結果と A_3 の最小公倍数を計算する。
- :
- 次に、前の計算結果と A_N の最小公倍数を計算する。

また、2 つの数 A, B には以下の性質が成り立ちます。（→2.5.2 項）

$$A \times B = (A \text{ と } B \text{ の最大公約数}) \times (A \text{ と } B \text{ の最小公倍数})$$

$$\text{すなわち} \quad (A \text{ と } B \text{ の最小公倍数}) = \frac{A \times B}{(A \text{ と } B \text{ の最大公約数})}$$

したがって、以下のようなプログラムを書くと、正解が得られます。なお、関数 $\text{LCM}(A, B)$ は A と B の最小公倍数を計算する関数です。

```

#include <iostream>
#include <algorithm>
using namespace std;

long long GCD(long long A, long long B) {
    while (A >= 1 && B >= 1) {
        if (A < B) B = B % A; // A < B の場合、大きい方 B を書き換える
        else A = A % B; // A >= B の場合、大きい方 A を書き換える
    }
}

```

次ページへ

```

    }
    if (A >= 1) return A;
    return B;
}

long long LCM(long long A, long long B) {
    return (A / GCD(A, B)) * B;
}

long long N;
long long A[100009];

int main() {
    // 入力
    cin >> N;
    for (int i = 1; i <= N; i++) cin >> A[i];

    // 答えを求める
    long long R = LCM(A[1], A[2]);
    for (int i = 3; i <= N; i++) {
        R = LCM(R, A[i]);
    }

    // 出力
    cout << R << endl;
    return 0;
}

```

※ Python などのソースコードは GitHub の codes フォルダをご覧ください。