

LFSR Challenge 1

Problem: Need to encrypt/decrypt 1000 characters

Will an 11-bit LFSR do the job?

A. Yes, no problem

B. No, the bits it produces are not random

C. No, need a longer LFSR

LFSR Challenge 2

Problem: Need to encrypt/decrypt a 1 gigabyte movie

How long a LFSR is needed?

A. 30 bits should be enough

B. 100 bits is safe

C. 1000 bits is definitely enough

If/While Challenge 1

Anything wrong with the following code?

```
double rate = 0.35;  
if (income < 47450) rate = 0.22;  
if (income < 114650) rate = 0.25;  
if (income < 174700) rate = 0.28;  
if (income < 311950) rate = 0.33;
```

If/While Challenge 2

Anything wrong with the following code?

```
public class PowersOfTwo {
    public static void main(String[] args) {
        int N = Integer.parseInt(args[0]);
        int i = 0; // loop control counter
        int v = 1; // current power of two
        while (i <= N)
            System.out.println(v);
            i = i + 1;
            v = 2 * v;
    }
}
```

Array Challenge 1

The following code sets array values to the 52 card values and prints them.
What order are they printed?

```
String[] deck = new String[52];  
for (int i = 0; i < 13; i++)  
    for (int j = 0; j < 4; j++)  
        deck[4*i + j] = rank[i] + " of " + suit[j];  
  
for (int i = 0; i < 52; i++)  
    System.out.println(deck[i]);
```

A. two of clubs
two of diamonds
two of hearts
two of spades
three of clubs
...

B. two of clubs
three of clubs
four of clubs
five of clubs
six of clubs
...

Array Challenge 2

How many multiplications to multiply two N-by-N matrices?

```
double[][] c = new double[N][N];  
for (int i = 0; i < N; i++)  
    for (int j = 0; j < N; j++)  
        for (int k = 0; k < N; k++)  
            c[i][j] += a[i][k] * b[k][j];
```

A. N

B. N^2

C. N^3

D. N^4

Functions Challenge 1.1

What happens when you compile and run the following code?

```
public class Cubes1
{
    public static int cube(int i)
    {
        int j = i * i * i;
        return j;
    }
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        for (int i = 1; i <= N; i++)
            StdOut.println(i + " " + cube(i));
    }
}
```

Functions Challenge 1.2

What happens when you compile and run the following code?

```
public class Cubes2
{
    public static int cube(int i)
    {
        int i = i * i * i;
        return i;
    }
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        for (int i = 1; i <= N; i++)
            StdOut.println(i + " " + cube(i));
    }
}
```


Functions Challenge 1.3

What happens when you compile and run the following code?

```
public class Cubes3
{
    public static int cube(int i)
    {
        i = i * i * i;
    }
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        for (int i = 1; i <= N; i++)
            StdOut.println(i + " " + cube(i));
    }
}
```

Functions Challenge 1.4

What happens when you compile and run the following code?

```
public class Cubes4
{
    public static int cube(int i)
    {
        i = i * i * i;
        return i;
    }
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        for (int i = 1; i <= N; i++)
            StdOut.println(i + " " + cube(i));
    }
}
```

Functions Challenge 1.5

What happens when you compile and run the following code?

```
public class Cubes5
{
    public static int cube(int i)
    {
        return i * i * i;
    }
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        for (int i = 1; i <= N; i++)
            StdOut.println(i + " " + cube(i));
    }
}
```

Functions Challenge 2

What sound does the following program produce?

```
public static double[] tone(double hz, double seconds)
{
    int SAMPLE_RATE = 44100;
    int N = (int) (seconds * SAMPLE_RATE);
    double[] a = new double[N+1];
    for (int i = 0; i <= N; i++)
        a[i] = Math.Random();
    return a;
}
```

Recursion Challenge 1.1 (difficult but important)

Is this an efficient way to compute $F(50)$?

```
public static long F(int n)
{
    if (n == 0) return 0;
    if (n == 1) return 1;
    return F(n-1) + F(n-2);
}
```

Recursion Challenge 1.2 (easy and also important)

Is this an efficient way to compute $F(50)$?

```
long[] F = new long[51];  
F[0] = 0; F[1] = 1;  
if (n == 1) return 1;  
for (int i = 2; i <= 50; i++)  
    F[i] = F[i-1] + F[i-2];
```

Performance Challenge 1

Let $F(N)$ be the running time of program `Mystery` for input N .

```
public static Mystery
{
    ...
    int N = Integer.parseInt(args[0]);
    ...
}
```

Observation: $F(2N)/F(N)$ is about 4.

What is the order of growth of the running time?

Performance Challenge 2

Let $F(N)$ be the running time of program `Mystery` for input N .

```
public static Mystery
{
    ...
    int N = Integer.parseInt(args[0]);
    ...
}
```

Observation: $F(2N)/F(N)$ is about 2.

What is the order of growth of the running time?

Performance Challenge 3

Is this an efficient way to compute binomial coefficients?

```
public static long binomial(long n, long k)
{
    if (k == 0) return 1;
    if (n == 0) return 0;
    return binomial(n-1, k-1) + binomial(n-1, k);
}
```

Performance Challenge 4

Let $F(N)$ be the time to compute $\text{binomial}(2N, N)$ using the naive algorithm.

```
public static long binomial(long n, long k)
{
    if (k == 0) return 1;
    if (n == 0) return 0;
    return binomial(n-1, k-1) + binomial(n-1, k);
}
```

Observation: $F(N+1)/F(N)$ is about 4.

What is the order of growth of the running time?

Performance Challenge 5

Let $F(N)$ be the time to compute $\text{binomial}(2N, N)$ using dynamic programming.

```
for (int n = 1; n <= 2*N; n++)  
    for (int k = 1; k <= N; k++)  
        bin[n][k] = bin[n-1][k-1] + bin[n-1][k];
```

What is the order of growth of the running time?

Performance Challenge 6

How much memory does this program use (as a function of N)?

```
public class RandomWalk
{
    public static void main(String[] args)
    {
        int N = Integer.parseInt(args[0]);
        int[][] count = new int[N][N];
        int x = N/2;
        int y = N/2;

        for (int i = 0; i < N; i++) {
            // no new variable declared in loop
            ...
            count[x][y]++;
        }
    }
}
```

Image Processing Challenge 1

What does the following code do? (Easy question!)

```
Picture pic = new Picture(args[0]);  
for (int i = 0; i < pic.width(); i++)  
    for (int j = 0; j < pic.height(); j++)  
        pic.set(i, j, pic.get(i, j)); pic.show();
```

Image Processing Challenge 2

What does the following code do? (Hard question.)

```
Picture pic = new Picture(args[0]);  
for (int i = 0; i < pic.width(); i++)  
    for (int j = 0; j < pic.height(); j++)  
        pic.set(i, pic.height()-j-1, pic.get(i, j));  
pic.show();
```

Image Processing Challenge 3

What does the following code do?

```
Picture source = new Picture(args[0]);
int width  = source.width();
int height = source.height();
Picture target = new Picture(width, height);
for (int i = 0; i < width; i++)
    for (int j = 0; j < height; j++)
        target.set(i, height-j-1, source.get(i, j));
target.show();
```

Search Challenge 1

A credit card company needs to whitelist 10 million customer accounts, processing 1000 transactions per second.

Using **sequential search**, what kind of computer is needed?

- A. Toaster
- B. Cellphone
- C. Your laptop
- D. Supercomputer
- E. Google server farm

Search Challenge 2

A credit card company needs to whitelist 10 million customer accounts, processing 1 thousand transactions per second.

Using **binary search**, what kind of computer is needed?

- A. Toaster
- B. Cellphone
- C. Your laptop
- D. Supercomputer
- E. Google server farm

Sort Challenge 1

A credit card company uses insertion sort to sort 10 million customer account numbers, for use in whitelisting with binary search. What kind of computer is needed?

- A. Toaster
- B. Cellphone
- C. Your laptop
- D. Supercomputer
- E. Google server farm

Sort Challenge 2

A credit card company uses **mergesort** to sort 10 million customer account numbers, for use in whitelisting with binary search. What kind of computer is needed?

- A. Toaster
- B. Cellphone
- C. Your laptop
- D. Supercomputer
- E. Google server farm

Sort Challenge 3

Four researchers A, B, C and D are looking for long repeated subsequences in a genome with over 1 billion characters.

- A has a grad student do it.
- B uses **brute force** (check all pairs) solution.
- C uses sorting solution with **insertion sort**.
- D uses sorting solution with **mergesort**

Which one is more likely to find a cancer cure?

Stack Challenge: Stack Sort?

Q. Can we always insert pop commands (-) to make strings come out sorted?

Ex 1: 6 5 4 3 2 1 - - - - -

Ex 2: 1 - 2 - 3 - 4 - 5 - 6 -

Ex 3: 4 1 - 3 2 - - - 6 5 - -

List-Processing Challenge 1

What does the following code do?

```
...  
Node list = null;  
while (!StdIn.isEmpty())  
{  
    Node old = list;  
    list = new Node();  
    list.item = StdIn.readString();  
    list.next = old;  
}  
for (Node t = list; t != null; t = t.next)  
    StdOut.println(t.item);  
...
```

List-Processing Challenge 2

What does the following code do?

```
...  
Node list = new Node();  
list.item = StdIn.readString();  
Node last = list;  
while (!StdIn.isEmpty())  
{  
    last.next = new Node();  
    last = last.next;  
    last.item = StdIn.readString();  
}  
...
```

No more challenges