Problem 1. Solve a Recurrence Relation

Find an explicit (closed) form for the sequence satisfying the:
$d_k = 4d_{k-2}, \forall k \in \mathbb{Z}, k \geq 2$.
$d_0 = 1, d_1 = -1$.

Problem 2. Solve a Recurrence Relation

Find an explicit (closed) form for the sequence satisfying the:
$r_k = 2r_{k-1} - r_{k-2}, \forall k \in \mathbb{Z}, k \geq 2.$
$r_0 = 1, r_1 = 4.$

Solution

Problem 3. Guess and Check

Guess the formula for the sequence $a_k = ka_{k-1}, \forall k \in \mathbb{Z} \geq 1, a_0 = 1$ by writing out the first few elements in the sequence.
Prove your guess with mathematical induction.

Problem 4. Guess and Check

Guess the formula for the sequence $b_k = b_{k-1} + 2k, \forall k \in \mathbb{Z} \geq 1, b_0 = 3$ by writing out the first few elements in the sequence.
Prove your guess with mathematical induction.

Problem 5. Asymptotic Notation

For each function $f(x)$ below give the slowest growing function $g(x)$ such that $f(x)$ is $O(g(x))$

- $f(x) = 7x^2 + 12x$
- $f(x) = 100x^5 - 50x^3 + 12x$

Suppose a computer takes 1 µs (microsecond) to execute an operation. How long will it take to execute the n_k operations. Convert your answer into seconds, minutes, hours, days, weeks, months, years and so on as appropriate. Use $n_0 = 10^2, n_1 = 10^4, n_2 = 10^8$ and $n_3 = 10^{16}$.

- $\log_2(n_k), 0 \leq k < 4$, i.e. a log-time algorithm.
- $n_k, 0 \leq k < 4$, i.e. a linear-time algorithm.
- $(n_k)^2, 0 \leq k < 4$, i.e. an n-squared algorithm.
- $2^{n_k}, 0 \leq k < 4$, i.e. an exponential algorithm.

Problem 6. Algorithm Running Time

Find the asymptotic running time of the following algorithm:

```
function Alg1
    i ← 1
    for i < n do
        j ← 1
        for j < 2n do
            a = 2 · n + i · j
        end for
    end for
    return a
end function
```

Problem 7. Algorithm Running Time

Find the asymptotic running time of the following algorithm:

```
function Fib(n)
    if n < 2 then
        return 1
    end if
    return Fib(n-1)+Fib(n-2)
end function
```

Problem 8. Algorithm Running Time

Find the asymptotic running time of the following algorithm:

```
function Fib(n)
    a = b = 1
    i ← 0
    for i < n do
        a ← a + b
        b ← a
    end for
    return a
end function
```

Problem 9. Algorithm Running Time

Find the asymptotic running time of the following algorithm:

```
Array[n] fibarray;
function Fib(n)
    return FibHelper(n);
end function
function FibHelper(n)
    if n ≤ 2 then
        return 1;
    end if
    if fibarray[n-1]=0 then
        fibarray[n-1]= FibHelper(n-1)
    end if
    if fibarray[n-2]=0 then
        fibarray[n-2]= FibHelper(n-2)
    end if
    return fibarray[n-1]+fibarray[n-2];
end function
```

Problem 10. Algorithm Running Time

Find the asymptotic running time of the following algorithm:

---

function FIB(n):
  $C \leftarrow \frac{1+\sqrt{5}}{2}$
  return $\frac{C^n-(-C)^n)}{2C}$
end function
