# ALGORITHMIC

hello, world

## 1. Basic forms

### 1.1. The Simple Statement.

$S \leftarrow O$

$S \leftarrow O$ {comment}

### 1.2. The body.

**do** {comment}
    something 1
    something 2

### 1.3. The *if-then-else* Statement.

**if** some condition is true **then** {comment}
    do some processing
**else if** some other condition is true **then** {comment}
    do some different processing
**else if** some even more bizarre condition is met **then** {comment}
    do something else
**else** {comment}
    do the default actions
**end if**

### 1.4. The *for* Loop.

**for** $i = 0$ to 10 **do** {comment}
    carry out some processing
**end for**

**for all** $i$ such that $0 \leq i \leq 10$ **do** {comment}
    carry out some processing
**end for**

**for** $i = 0$ **to** 10 **do** {comment}
    carry out some processing
**end for**

### 1.5. The *while* Loop.

**while** some condition holds **do** {comment}
    carry out some processing
**end while**

### 1.6. The *repeat-until* Loop.

**repeat** {comment}
    carry out some processing
**until** some condition is met

1.7. **The Infinite Loop.**

```
loop {comment}
    this processing will be repeated forever
end loop
```

1.8. **The Precondition.**

**Require:** $x \neq 0$ and $n \geq 0$

1.9. **The Postcondition.**

**Ensure:** $x \neq 0$ and $n \geq 0$

1.10. **Globals.**

```
globals x, y
```

1.11. **Inputs.**

```
inputs  {comment}
    x, y
```

1.12. **Outputs.**

```
outputs  {comment}
    x, y
```

1.13. **Returning Values.**

```
return  (x + y)/2
```

1.14. **Printing Messages.**

```
print ``Hello, World!''
```

1.15. **Comments.**

```
do something {this is a comment}
```

## 2. Some longer examples

```
a ← 1
if a is even then
    print  "a is even"
else if a is odd then
    print  "a is odd"
else
    print  "a is really weird"
end if
```

**Require:** $n \geq 0$
**Ensure:** $y = x^n$

```
y ← 1
X ← x
N ← n
while N ≠ 0 do
    if N is even then
        X ← X × X
        N ← N/2
    else {N is odd}
```

$y \leftarrow y \times X$
$N \leftarrow N - 1$
**end if**
**end while**

2.1. **mcom3655.**

**Input:** Choose an arbitrary $\alpha_L^0 \in \mathbb{C}^{(L+1)^2}$, $\{\epsilon^k\}$ with $\epsilon^k > 0$. Set $k = 0$.
**while** a termination criterion is not met, **do**
1) Solve the weighted $\ell_2$ minimization problem

$$\alpha_{l\cdot}^{k+1} = \arg\min_{\alpha_{l\cdot}\in\mathbb{C}^{2l+1}} \left\{ \frac{1}{2}\|\alpha_{l\cdot} - \alpha_{l\cdot}^{\circ}\|_2^2 + \frac{1}{2}\lambda p \beta_l w_l^k \|\alpha_{l\cdot}\|_2^2 \right\}, l = 0, 1, \ldots, L, \tag{1}$$

where $w_l^k = (\|\alpha_{l\cdot}^k\|_2^2 + \epsilon^k)^{\frac{p}{2}-1}$.
2) Set $k \leftarrow k + 1$ and go to step 1).
**end while**

2.2. **mcom3356.**

**Require:** A lower trapezoidal $n \times (n-1)$ matrix $\mathbf{H} = (h_{i,j})$ with $h_{i,j} = 0$ if $j > i$ and $h_{j,j} \neq 0$.
**Ensure:** A unimodular matrix $\mathbf{D}$ such that $\mathbf{H} := \mathbf{D} \cdot \mathbf{H} = (h_{i,j})$ satisfying $|h_{i,j}| \leq |h_{j,j}|/2$ for $1 \leq j < i \leq n$.
1: $\mathbf{D} := \mathbf{I}_n$.
2: **for** $i$ from 2 to $n$ **do**
3: **for** $j$ from $i-1$ to 1 by stepsize $-1$ **do**
4: $q := \lfloor h_{i,j}/h_{j,j} + 0.5 \rfloor$.
5: **for** $k$ from 1 to $n$ **do**
6: $d_{i,k} := d_{i,k} - q d_{j,k}$.
7: **end for**
8: **end for**
9: **end for**

2.3. **mcom3363.**

**Require:** A polynomial $h$ with irreducible factors of degree $d$ over $k = \mathbb{F}_q[X]/f(X)$.
**Ensure:** An irreducible factor of $h$ over $k$.
1: If $\deg h = d$ return $h$.
2: Take a random polynomial $a_0 \in k[Z]$ of degree less than $\deg h$,
3: Compute $a_1 \leftarrow \sum_{i=0}^{md-1} a_0^{q^i} \mod h$,
4: **if** $q$ is an even power $q = 2^e$ **then**
5: Compute $a_2 \leftarrow \sum_{i=0}^{e-1} a_1^{2^i} \mod h$
6: **else**
7: Compute $a_2 \leftarrow a_1^{(q-1)/2} \mod h$
8: **end if**
9: Compute $h_0 \leftarrow \gcd(a_2, h)$ and $h_1 \leftarrow \gcd(a_2 - 1, h)$ and $h_{-1} \leftarrow h/(h_0 h_1)$,
10: Apply recursively to the smallest non-constant polynomial among $h_0, h_1, h_{-1}$.

2.4. **mcom3385 (with endtags=no).**

**Require:** A rational number $\alpha = a/b$.
**Ensure:** The algorithm tells whether the RCF expansion of $\alpha$ is finite or periodic.
1: $x := \alpha$

2: $B_1 := \max\left(\frac{\log b}{\log \ell}, 2\right)$
3: **for** $i = 1$ to $B_1$ **do**
4:  **if** $x < 0$ **then**
5:  **return** The expansion is periodic.
6:  $y := x - \lfloor x \rfloor_\ell$
7:  **if** $y == 0$ **then**
8:  **return** The expansion is finite.
9:  $x := 1/y$