

## Algoritmi

---

**Algorithm 1** Algoritmul lui Euclid

---

```
1: procedure EUCLID( $a, b$ ) ▷ C.m.m.d.c al lui a si b
2:    $r \leftarrow a \bmod b$ 
3:   while  $r \neq 0$  do ▷ Avem raspunsul daca  $r = 0$ 
4:      $a \leftarrow b$ 
5:      $b \leftarrow r$ 
6:      $r \leftarrow a \bmod b$ 
7:   return  $b$  ▷ C.m.m.d.c este b
```

---

---

**Algorithm 2** Algoritmi elementari de sortare

---

```
1: procedure INSERT( $T[1..n]$ )
2:   for  $i \leftarrow 2$  to  $n$  do
3:      $x \leftarrow T[i]; j \leftarrow i - 1$ 
4:     while  $j > 0$  and  $x < T[j]$  do
5:        $T[j + 1] \leftarrow T[j]$ 
6:        $j \leftarrow j - 1$ 
7:      $T[j + 1] \leftarrow x$ 
8: procedure SELECT( $T[1..n]$ )
9:   for  $i \leftarrow 1$  to  $n - 1$  do
10:     $minj \leftarrow i; minx \leftarrow T[i]$ 
11:    for  $j \leftarrow i + 1$  to  $n$  do
12:      if  $T[j] < minx$  then
13:         $minj \leftarrow j$ 
14:         $minx \leftarrow T[j]$ 
15:     $T[minj] \leftarrow T[i]$ 
16:     $T[i] \leftarrow T[minx]$ 
```

---

---

**Algorithm 3** Algoritm Greedy

---

```
1: procedure GREEDY( $C$ ) ▷  $C$  este mulțimea candidaților
2:    $S \leftarrow \emptyset$  ▷  $S$  este mulțimea în care construim soluția
3:   while not soluție( $S$ ) and  $C \neq \emptyset$  do
4:      $x \leftarrow$  un element din  $C$  care maximizează/minimizează select( $x$ )
5:      $C \leftarrow C \setminus \{x\}$ 
6:     if fezabil( $S \cup \{x\}$ ) then
7:        $S \leftarrow S \cup \{x\}$ 
8:   if soluție( $S$ ) then
9:     return  $S$ 
10:  else
11:    return "nu există soluție"
```

---

---

**Algorithm 4** Algoritmul înmulțirii "a la russe"

---

```
1: procedure RUSSE( $A, B$ )
2:   arrays  $X, Y$ 
3:    $X[1] \leftarrow A; Y[1] \leftarrow B$  ▷ inițializare
4:    $i \leftarrow 1$  ▷ se construiesc cele două coloane
5:   while  $X[i] > 1$  do
6:      $X[i+1] \leftarrow X[i] \text{ div } 2$  ▷ div reprezintă împărțirea întreagă
7:      $Y[i+1] \leftarrow Y[i] + Y[i]$ 
8:      $i \leftarrow i + 1$  ▷ adună numerele  $Y[i]$  corespunzătoare numerelor  $X[i]$  impare
9:    $prod \leftarrow 0$ 
10:  while  $i > 0$  do
11:    if  $X[i]$  este impar then
12:       $prod \leftarrow prod + Y[i]$ 
13:       $i \leftarrow i - 1$ 
14:  return  $prod$ 
```

---

---

**Algorithm 5** Șirul lui Fibonacci

---

```
1: procedure FIB3( $n$ )
2:    $i \leftarrow 1; j \leftarrow 0; k \leftarrow 0; h \leftarrow 1$ 
3:   while  $n > 0$  do
4:     if  $n$  este impar then
5:        $t \leftarrow jh$ 
6:        $j \leftarrow ih + jk + t$ 
7:        $i \leftarrow ik + t$ 
8:        $t \leftarrow h^2$ 
9:        $h \leftarrow 2kh + t$ 
10:       $k \leftarrow k^2 + t$ 
11:       $n \leftarrow n \text{ div } 2$ 
12:  return  $j$ 
```

---