



FIGURE 1 – De gauche à droite : écriture de l’algorithme d’Euclide en langage C, Python et OCaml sous emacs.

```

1 #include <stdio.h>
2
3 int main(void)
4 {
5     int a = 1;
6     if (a > 3)
7     {
8         printf("a est plus grand que 3\n");
9     }
10    else if (a > 2) // cas alternatif facultatif
11    {
12        printf("a est plus grand que 2\n");
13    }
14    else
15    {
16        printf("a est plus petit que 2\n");
17    }
18    return 0;
19 }
20

```

### Langage C

```

1
2
3
4
5 a = 1
6 if (a > 3) :
7
8     print("a est plus grand que 3")
9
10 elif (a > 2):
11
12     print("a est plus grand que 2")
13
14 else:
15
16     print("a est plus petit que 2")
17

```

### Langage Python

```

1
2
3
4
5 let a = 1 in
6 if (a > 3) then
7
8     Printf.printf "a est plus grand que 3\n"
9
10 else if (a > 2) then
11
12     Printf.printf "a est plus grand que 2\n"
13
14 else
15
16     Printf.printf "a est plus petit que 2\n"
17
18 ;;
19

```

### Langage OCaml

FIGURE 5 – Branchements conditionnels dans les trois langages C, Python et OCaml.

<pre> 1 #include &lt;stdio.h&gt; 2 3 int main(void) 4 { 5     unsigned int i; 6     float sum = 1.5; 7     for (i = 0; i &lt; 10; i++) 8     { 9         sum = sum + 2.5; 10        printf("Iteration %d : sum contient %f\n", i, sum); 11    } 12 13    return 0; 14 } 15 </pre>	<pre> 1 2 3 4 5 6 sum = 1.5 7 8 for i in range(0, 10, 1): 9 10     sum = sum + 2.5 11     print("Iteration ", i, ": sum contient", sum); </pre>	<pre> 1 2 3 4 5 6 let sum = ref 1.5 in 7 8   for i = 0 to 10 9   do 10     sum := !sum +. 2.5; 11     Printf.printf "Iteration %d : sum contient %f\n" i !sum; 12   done 13 14 ;; 15 </pre>
Language C	Language Python	Language OCaml

FIGURE 6 – Boucle for dans les trois langages C, Python et OCaml.