

# Nested loops and if conditions

Created using Maple 14.01

Jake Bobowski

```
> restart;
with(StringTools) :
FormatTime("%m-%d-%Y, %H:%M");
"08-04-2012, 07:12"
```

(1)

Here is an **if** condition inside of a **for** loop inside of a **while** loop. Nested structures very quickly become complicated and hard to decipher. It is a very good idea to indent your code so that it is easier to read. The only way that I've found to indent code is by inserting single spaces one at a time using the space bar. If you find a better way, let me know...

```
> i := 0 :
while i ≤ 20 do;
  x := mod(i, 2) :
  for j from 1 to 3 by 1 do;
    if x = 0 and j = 3 then;
      printf("%g%s%g%s", i, ` `, j, ` `);
    end if;
  end do;
  i := i + 1 :
end do;
0.3 2.3 4.3 6.3 8.3 10.3 12.3 14.3 16.3 18.3 20.3
```

Here's a nested structure that will list the first  $N$  prime numbers starting from (and including)  $num$ .

```
> N := 5 :
num := 2 :
i := 0 :
while i < N do;
  x := 2;
  while mod(num, x) ≠ 0 and x < num do;
    x := x + 1;
  end do;
  if x = num then;
    print(num);
    i := i + 1;
  end if;
  num := num + 1;
end do;
2
3
5
7
11
```

(2)

Let's make  $N$  and  $num$  more interesting values.

```
> N := 100 :
num := 10000 :
```

```
i := 0 ;
while i < N do;
  x := 2;
  while mod(num, x) ≠ 0 and x < num do;
    x := x + 1;
  end do;
  if x = num then;
    print(num);
    i := i + 1;
  end if;
  num := num + 1;
end do;
```

10007  
10009  
10037  
10039  
10061  
10067  
10069  
10079  
10091  
10093  
10099  
10103  
10111  
10133  
10139  
10141  
10151  
10159  
10163  
10169  
10177  
10181  
10193  
10211  
10223  
10243  
10247  
10253  
10259  
10267  
10271  
10273  
10289

10301  
10303  
10313  
10321  
10331  
10333  
10337  
10343  
10357  
10369  
10391  
10399  
10427  
10429  
10433  
10453  
10457  
10459  
10463  
10477  
10487  
10499  
10501  
10513  
10529  
10531  
10559  
10567  
10589  
10597  
10601  
10607  
10613  
10627  
10631  
10639  
10651  
10657  
10663  
10667  
10687  
10691

10709  
10711  
10723  
10729  
10733  
10739  
10753  
10771  
10781  
10789  
10799  
10831  
10837  
10847  
10853  
10859  
10861  
10867  
10883  
10889  
10891  
10903  
10909  
10937  
10939

