Repetition in Turing

Nested Counted Loops Accumulating Values

Accumulating Values

In many applications, we want to find the total amount of some quantity.

For example:

- adding student grades
- cash register totals
- scoring a game

Sample Code – Adding 3 Numbers

var num, sum : int sum := 0 % initialize sum to zero put "This program will add 3 integers" for count : 1 ... 3 put "Enter number ", count, ": " .. get num sum := sum + num end for

put "The sum is ", sum

Write a program that will output the values "10 20 30" on the screen.
Have this pattern repeat 5 times.
Sample Output: 10 20 30 30 10 20

30

Consider a simple FOR loop, which counts from 10 to 30:

for j : 10 .. 30 by 10
 % j will be 10, 20, 30
 put j
end for

We could duplicate this code multiple times, which would produce the desired effect:

```
for j : 10 .. 30 by 10
 % j will be 10, 20, 30
 put j
end for
for j : 10 .. 30 by 10
 % j will be 10, 20, 30
 put j
end for
for j : 10 .. 30 by 10
  % j will be 10, 20, 30
 put j
end for
```

We could duplicate this code multiple times, which would produce the desired effect:

for pu end	it :	j	10		30	by	10 -
for pu end	it :	j	10	•••	30	by	10
for pu end	it :	j	10	•••	30	by	10

This is just the same code, repeated multiple times.

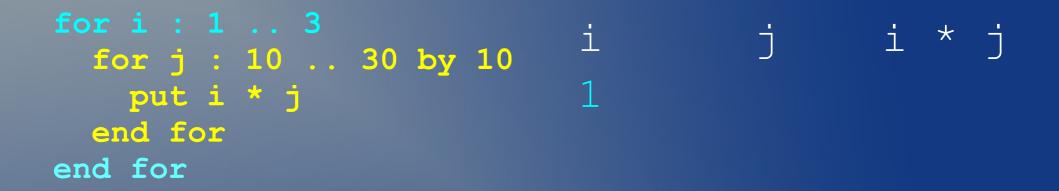
To repeat identical code, we use loops

We could duplicate this code multiple times, which would produce the desired effect:

for j : put j end for	10		30 :	by	10	
for j : put j end for	10	•••	30 :	by	10	<pre>for i : 1 3 for j : 10 30 by 10 put j end for end for</pre>
for j : put j end for	10	•••	30	by	10	

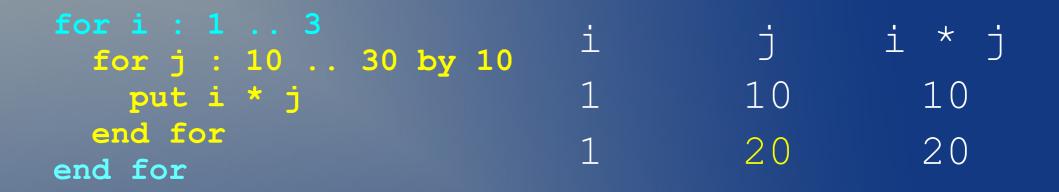
Our original problem statement wanted to output the sequence "10 20 30" five times, so we need a small adjustment to our solution:

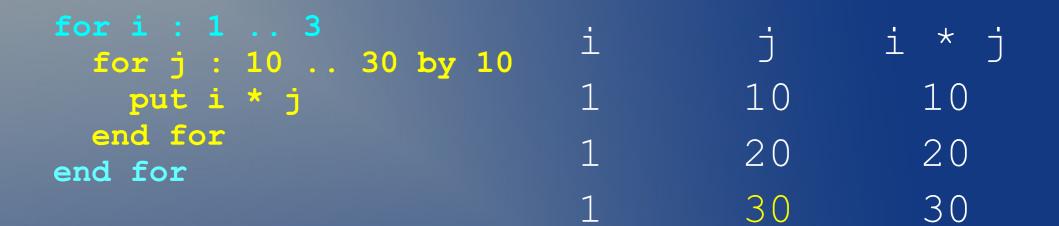
for i : 1 ... 5
 for j : 10 ... 30 by 10
 put j
 end for
end for

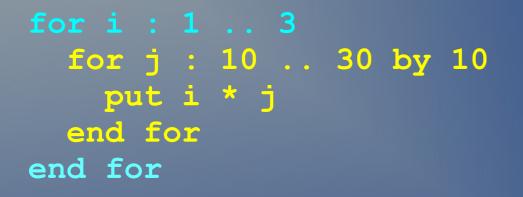


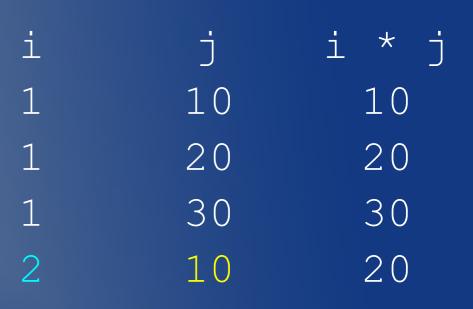


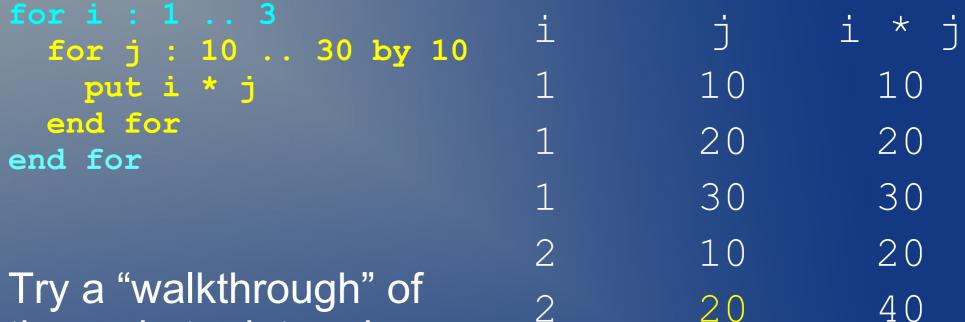


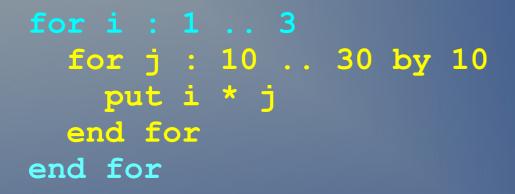












i	j	i * j
1	10	10
1	20	20
1	30	30
2	10	20
2	20	40
2	30	60
3	10	30
3	20	60
3	30	90