COMP108
Data Structures and Algorithms

Pseudo code (Part II)

Professor Prudence Wong

pwong@liverpool.ac.uk

2024-25

Developing pseudo code

- Suppose x & y are both +ve integers.
- Write a while loop to output all factors of x which are not factors of y.
- Examples of expected results:

x	y	factors of x	output
6	3	1, 2, 3, 6	2, 6
30	9	1, 2, 3, 5, 6, 10, 15, 30	2, 5, 6, 10, 15, 30
3	6	1, 3	-

- Skeleton:

```
i ← ?
while ???? do
begin
   if ??? then
      output ???
   i ← ??
end
```

Factors of x but not factors of y

Steps:

- Factor of x must be between 1 and x

```
i ← 1
while i ≤ x do
begin
   i ← i+1
end
```

- If x is divisible by i, then i is a factor of x

```
if x%i == 0 then
```

- If y is not divisible by i, then i is not a factor of y

```
if y%i ≠ 0 then
```

- We need both conditions to hold and then we output i

```
if x%i == 0 AND y%i ≠ 0 then
   output i
```

Factors of x but not factors of y

Summarizing:

```
i ← 1
while i ≤ x do
begin
   if x%i == 0 AND y%i ≠ 0 then
      output i
   i ← i+1
end
```

Finding Lowest Common Multiple (LCM)

Suppose 0 < x < y & both are +ve integers.

Write a while loop to output the lowest common multiple lcm of x and y, i.e., the smallest number that is divisible by x and divisible by y.

- So we want
  - lcm%x to be 0
  - lcm%y to be 0
  - lcm to be as small as possible
- lcm cannot be smaller than y and would not be larger than x*y.
  - we can start lcm from y, increase by 1 every time, and then check if lcm is multiple of both x and y
  - we can start lcm from y, increase by y every time, and then check if lcm is multiple of x (we don't need to check if lcm is multiple of y because it is already)
  - once we find the smallest one, we should stop

LCM

```
lcm ← __y__ , found ← false
while lcm ≤ __x*y__ AND found ≠ true do
begin
    if __lcm%x==0 AND lcm%y==0__ then
        found ← true
    else lcm ← lcm + __1__
end
output lcm
```

```
lcm ← __y__ , found ← false
while lcm ≤ __x*y__ AND found ≠ true do
begin
    if __lcm%x==0__ then
        found ← true
    else lcm ← lcm + __y__
end
output lcm
```

Questions

- Is the condition ``lcm ≤ x*y'' necessary?
- Why do we need to use the flag variable `found`?
- What happens if we remove the keyword ``else''?

More example

Suppose 0 < x < y & both are +ve integers.

Write a while loop to output all numbers each of which is

- a factor of x but not a factor of y, OR
- a factor of y but not a factor of x.

Examples of expected results:

x	y	factors of x	factors of y	output
3	6	1, 3	1, 2, 3, 6	2, 6
5	7	1, 5	1, 7	5, 7

x % i OR i % x ?

```
i ← 1
while i ≤ _____ do
begin
    if ____________________________________________ then
        output i
    i ← i+1
end
```

More example

Suppose 0 < x < y & both are +ve integers.

Write a while loop to output all numbers each of which is

- a factor of x but not a factor of y, OR
- a factor of y but not a factor of x.

Examples of expected results:

x	y	factors of x	factors of y	output
3	6	1, 3	1, 2, 3, 6	2, 6
5	7	1, 5	1, 7	5, 7

```
i ← 1
while i ≤ ____y____ do
begin
    if ____(x%i==0 AND y%i ≠0)____ OR ____(y%i==0 AND x%i ≠0)____ then
        output i
    i ← i+1
end
```

More example

Suppose 0 < x < y & both are +ve integers. Write a while loop to output all numbers each of which is

- a factor of x but not a factor of y, OR
- a factor of y but not a factor of x.

factor of x	factor of y	output?
T	T	F
T	F	T
F	T	T
F	F	F

Do you remember what this is? It's **XOR**.

```
i ← 1
while i ≤ y do
begin
    if x%i==0 XOR y%i==0 then
        output i
    i ← i+1
end
```

Summary: Developing pseudo code

More Exercises on pseudo code in Weekly Practice Quiz and next week's tutorial

Next: Algorithm Efficiency, Use of Arrays

For note taking