Sheet 1 - Sol

You need to keep an eye on the formal definition of algorithm:

"An algorithm is an ordered set of unambiguous, executable steps that defines a terminating process."

1

• 5.4
  Let the students come up with their own examples from whatever domain they prefer.

• 5.5
  No, it does not represent an algorithm in the strict sense.
  Because the process described will never terminate as the value of `Count` will never be 5.

• 5.6
  The three steps do not constitute an algorithm because Step 3 is not executable as the two line segments drawn in the two previous steps do not intersect.

• 5.7

```plaintext
Count ← 2;
repeat {
  print Count;
  Count ← Count + 1;
}
until (Count ≥ 7)
```

• 5.13
  Pseudocode is a relaxed version of a programming language used to jot down ideas. A formal programming language prescribes strict rules of grammar that must be obeyed.

• 5.27
  Identify the termination condition in each of the following iterative statements:
  a) `Count ≥ 5`
  b) `Count = 1`
  c) `(Count ≥ 5) or (Total ≥ 56)`

• 5.28
  The body of the loop is `{print Count; Count ← Count + 3;}` and it will be executed twice.
  If the test is changed to `(Count ≠ 6)`, the body will be executed infinitely.
II

Given
Count ← 0;
while (Count < 10) do {
    print Count;
    Count ← Count + 1;
}

a) Count ← 0;
while (Count < 10) do {
    print 9 - Count;
    Count ← Count + 1;
}

b) Count ← 0;
while (Count < 10) do {
    print Count;
    Count ← Count + 2;
}

c) Count ← 1;
while (Count < 10) do {
    print Count;
    Count ← Count + 2;
}

d) Count ← 0;
while (Count < 10) do {
    print "*";
    Count ← Count + 1;
}

e) Sum ← 0;
Count ← 0;
while (Count < 10) do {
    Sum ← Sum + Count;
    Count ← Count + 1;
} 
print Sum;