



## Sheet 1

I Solve the following *Review Problems* from *Computer Science: An Overview*:

• **5.4**

Select a subject with which you are familiar and design a pseudocode for giving directions in that subject. In particular, describe the primitives you would use and the syntax you would use to represent them. (If you are having trouble thinking of a subject, try sports, arts, or crafts.)

• **5.5**

Does the following program represent an algorithm in the strict sense? Why or why not?

```
Count ← 0;
while (Count ≠ 5) do
    Count ← Count + 2;
```

• **5.6**

In what sense do the following three steps not constitute an algorithm?

```
Step 1: Draw a straight line segment between the points with rectangular coordinates (2,5) and (6,11).
Step 2: Draw a straight line segment between the points with rectangular coordinates (1,3) and (3,6).
Step 3: Draw a circle whose center is at the intersection of the previous line segments and radius is 2.
```

• **5.7**

Rewrite the following program using a **repeat** structure instead of a **while** structure. Be sure the new version prints the same values as the original.

```
Count ← 2;
while (Count < 7) do {
    print Count;
    Count ← Count + 1;
}
```

• **5.13**

What is the difference between a *formal programming language* and a *pseudocode*?

• **5.27**

Identify the termination condition in each of the following iterative statements:

a)

```
while (Count < 5) do
    { ... }
```

b)

```
repeat
    { ... }
until (Count = 1)
```

c)

```
while ((Count < 5) and (Total < 56)) do
    { ... }
```

• **5.28**

Identify the body of the following loop structure and count the number of times it will be executed. What happens if the test is changed to (Count ≠ 6)?

```
Count ← 1;
while (Count ≠ 7) do {
    print Count;
    Count ← Count + 3;
}
```



Benha University

# Computer ECE 001



Computer Systems Engineering  
Electrical Engineering Department

Faculty of Engineering  
(at Shoubra)

II The following program prints the numbers from 0 to 9:

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

Modify it to:

- Print the numbers from 9 down to 0
- Print the even numbers only from 0 to 9
- Print the odd numbers only from 0 to 9
- Print the character '\*' ten times
- Print the sum of the numbers from 0 to 9

Represent your answers using flowcharts as well as pseudocode.