1. A B C D E
2. A B C D E
3. A B C D E
4. A B C D E
5. A B C D E
6. A B C D E
7. A B C D E
8. A B C D E
9. A B C D E
10. A B C D E
11. A B C D E
12. A B C D E
13. A B C D E
14. A B C D E
15. A B C D E
16. A B C D E
17. A B C D E
18. A B C D E
19. A B C D E
20. A B C D E

From the following options:
A) Count ← 1;
while (Count ≠ 10) do {
    print Count;
    Count ← Count + 2;
}

B) Count ← 0;
while (Count ≠ 10) do {
    print Count;
    Count ← Count + 2;
}

C) Count ← 10;
while (Count ≥ 0) do {
    Count ← Count - 2;
    print 8 - Count;
}

D) Count ← 10;
repeat {
    print Count;
    Count ← Count - 2;
} until (Count > 0)

E) Count ← 0;
repeat {
    Count ← Count + 2;
    print Count;
} until (Count = 10)

Identify:
1. Two algorithms that have the same stop condition
2. Two programs that produce the same output

From the following options:
3. A non terminating program
4. Two programs that do the same number of iterations
5. A program that does only one iteration

From the following options:
A) Count ← 1;
while (Count ≠ 10) do {
    print Count;
    Count ← Count + 2;
}

B) Count ← 0;
while (Count ≠ 10) do {
    print Count;
    Count ← Count + 2;
}

C) Count ← 10;
while (Count ≥ 0) do {
    Count ← Count - 2;
    print 8 - Count;
}

D) Count ← 10;
repeat {
    print Count;
    Count ← Count - 2;
} until (Count > 0)

E) Count ← 0;
repeat {
    Count ← Count + 2;
    print Count;
} until (Count = 10)

Identify:
6. AND gate
7. NAND gate
8. NOR gate
9. XOR gate

From the following options:
A) 4 5/16
B) 4 3/16
C) 4 3/8
D) 5 3/8
E) 5 3/16

Identify:
10. Bit
11. Byte
12. Hexadecimal Digit
13. ASCII Code

From the following options:
A) 4 5/16
B) 4 3/16
C) 4 3/8
D) 5 3/8
E) 5 3/16

Identify:
14. 100.0101
15. 100.101
16. 101.011
17. 100.0011

From the following options:
A) 1110
B) 1011
C) 0110
D) 1001
E) 1101

Identify:
18. 0011 + 0011
19. 1100 + 0010
20. 1011 + 0010