Question 1

1. **For printf statements below, determine the output:**

c

Copy code

```
printf("%%8.2f = %8.2f\n", 123.456);
printf("%%-8.2f = %-8.2f\n", 123.456);
printf("%%10.3g = %10.3g\n", 0.00098654);
printf("%%-10.4g = %-10.4g\n", 0.00087654);
printf("%%.1f = %.1f\n", 45.6789);
```

2. **Answers**:
    a. %8.2f aligns to the right, width 8, two decimal places: 123.46
    b. %-8.2f aligns to the left, width 8, two decimal places: 123.46
    c. %10.3g gives 3 significant figures: 9.87e-04
    d. %-10.4g gives 4 significant figures, left aligned: 8.765e-04
    e. %.1f rounds to 1 decimal place: 45.7

Question 2

1. **Predict the output for each printf statement:**

c

Copy code

```
printf("%%.5e = %.5e\n", 12345.678);
printf("%%12.4e = %12.4e\n", 12345.678);
printf("%%-10.2e = %-10.2e\n", 12345.678);
printf("%%8.3f = %8.3f\n", 12.3456);
printf("%%-.1g = %-.1g\n", 0.0005678);
```

2. **Answers**:
    a. %.5e uses 5 decimal places: 1.23457e+04

b. %12.4e aligns right, width 12, four decimal places: 1.2346e+04

c. %-10.2e aligns left with width 10, two decimal places: 1.23e+04

d. %8.3f rounds to three decimal places, right aligned: 12.346

e. %-.1g with one significant figure: 6e-04

Question 3

1. **Output the following using various formats for integer and floating-point precision:**

c

Copy code

```
printf("%%7d = %7d\n", 123);
printf("%%-7d = %-7d\n", 123);
printf("%%07d = %07d\n", 123);
printf("%%5.2f = %5.2f\n", 12.3);
printf("%%5.0f = %5.0f\n", 9.8);
```

2. **Answers**:

a. %7d aligns right, width 7: 123

b. %-7d aligns left, width 7: 123

c. %07d pads with leading zeroes: 0000123

d. %5.2f gives 5 total width, 2 decimal places: 12.30

e. %5.0f rounds with no decimals: 10

Question 4

1. **Predict the output for hexadecimal and octal formats:**

c

Copy code

```
printf("%%5x = %5x\n", 255);
printf("%%#5x = %#5x\n", 255);
printf("%%5o = %5o\n", 16);
printf("%%#5o = %#5o\n", 16);
printf("%%#-6o = %#-6o\n", 16);
```

2. **Answers**:
    a. %5x outputs hexadecimal, right aligned: ff
    b. %#5x with prefix 0x: 0xff
    c. %5o outputs octal, right aligned: 20
    d. %#5o with prefix 0: 020
    e. %#-6o aligns left with prefix 0: 020

Question 5

1. **Output the following using scientific notation:**

c

Copy code

```
printf("%%6.2E = %6.2E\n", 9876.543);
printf("%%10.3E = %10.3E\n", 0.000567);
printf("%%.2E = %.2E\n", 1234.56);
printf("%%6.1E = %6.1E\n", 98765.43);
printf("%%12.5E = %12.5E\n", 0.004321);
```

2. **Answers**:
    a. %6.2E: 9.88E+03
    b. %10.3E: 5.670E-04
    c. %.2E: 1.23E+03
    d. %6.1E: 9.9E+04
    e. %12.5E: 4.32100E-03

Question 6

1. **Explore width, padding, and alignment with characters:**

c

Copy code

```
printf("%%3c = %3c\n", 'A');
printf("%%-3c = %-3c\n", 'B');
printf("%%3c = %3c\n", 'C');
printf("%%3s = %3s\n", "Hi");
printf("%%-5s = %-5s\n", "Hey");
```

2. **Answers**:

    a. %3c: A

    b. %-3c: B

    c. %3c: C

    d. %3s: Hi

    e. %-5s: Hey

Question 7

1. **Predict outputs for float formatting with various precision settings:**

c

Copy code

```
printf("%%.4f = %.4f\n", 3.14159);
printf("%%8.1f = %8.1f\n", 2.718);
printf("%%8.2f = %8.2f\n", 0.678);
printf("%%5.0f = %5.0f\n", 3.14);
printf("%%6.3f = %6.3f\n", 9.8765);
```

2. **Answers**:

    a. %.4f: 3.1416

    b. %8.1f: 2.7

c. %8.2f: 0.68

d. %5.0f: 3

e. %6.3f: 9.877

Question 8

1. **Test different padding in integer formatting:**

c

Copy code

```
printf("%%06d = %06d\n", 78);
printf("%%-6d = %-6d\n", 78);
printf("%%4.3d = %4.3d\n", 7);
printf("%%5.0d = %5.0d\n", 45);
printf("%%#05x = %#05x\n", 26);
```

2. **Answers**:

a. %06d: 000078

b. %-6d: 78

c. %4.3d: 007

d. %5.0d: 45

e. %#05x: 0x01a

Question 9

1. **Practice with percentage and escape characters:**

c

Copy code

```
printf("%%d %% = %d%%\n", 95);
printf("%%c = %c\n", '%');
printf("Price: $%%.2f\n", 9.99);
```

2. **Answers**:
    a. 95%
    b. %
    c. Price: $9.99

Question 10

1. **Mix precision and width in octal and hexadecimal formats:**

c

Copy code

```
printf("%%6o = %6o\n", 123);
printf("%%#8x = %#8x\n", 255);
printf("%%-#5x = %-#5x\n", 32);
printf("%%#-7o = %#-7o\n", 16);
printf("%%8o = %8o\n", 64);
```

2. **Answers**:
    a. %6o: 173
    b. %#8x: 0xff
    c. %-#5x: 0x20
    d. %#-7o: 020
    e. %8o: 64

Question 11

1. **Predict outputs for different integer and floating-point formats:**

c

Copy code

```
printf("%%10.1f = %10.1f\n", 123.45);
printf("%%6.0f = %6.0f\n", 9.8765);
printf("%%4d = %4d\n", 321);
printf("%%-5.3d = %-5.3d\n", 12);
printf("%%-6.2f = %-6.2f\n", 56.78);
printf("%%.1g = %.1g\n", 123456.78);
```

2. **Answers**:

a. %10.1f: 123.5
b. %6.0f: 10
c. %4d: 321
d. %-5.3d: 012
e. %-6.2f: 56.78
f. %.1g: 1e+05

Question 12

1. **Explore floating-point formats with different widths and precision:**

c

Copy code

```
printf("%%9.3g = %9.3g\n", 9876.5);
printf("%%7.2g = %7.2g\n", 0.00234);
printf("%%.1e = %.1e\n", 123.456);
printf("%%.4g = %.4g\n", 0.01234);
printf("%%-10.5e = %-10.5e\n", 0.004321);
printf("%%12.2e = %12.2e\n", 123456.789);
```

2. **Answers**:

a. %9.3g: 9.88e+03
b. %7.2g: 0.0023
c. %.1e: 1.2e+02
d. %.4g: 0.01234
e. %-10.5e: 4.32100e-03
f. %12.2e: 1.23e+05

Question 13

1. **Apply padding and alignment for integers and octal formats:**

c

Copy code

```
printf("%%05o = %05o\n", 27);
printf("%%#7x = %#7x\n", 31);
printf("%%-8d = %-8d\n", 75);
printf("%%6.4d = %6.4d\n", 5);
printf("%%#8o = %#8o\n", 15);
printf("%%#5x = %#5x\n", 255);
```

2. **Answers**:
   a. %05o: 00033
   b. %#7x: 0x1f
   c. %-8d: 75
   d. %6.4d: 0005
   e. %#8o: 017
   f. %#5x: 0xff

Question 14

1. **Predict outputs for complex combinations of width, precision, and scientific notation:**

c

Copy code

```
printf("%%.4e = %.4e\n", 5678.9);
printf("%%6.2f = %6.2f\n", 3.14159);
printf("%%10.5e = %10.5e\n", 12.345);
printf("%%-9.3f = %-9.3f\n", 987.65);
printf("%%7.3g = %7.3g\n", 56789.01);
printf("%%.3E = %.3E\n", 0.000123);
```

2. **Answers**:
    a. %.4e: 5.6789e+03
    b. %6.2f: 3.14
    c. %10.5e: 1.23450e+01
    d. %-9.3f: 987.650
    e. %7.3g: 5.68e+04
    f. %.3E: 1.230E-04

Question 15

1. **Format floats and integers with left and right alignment:**

c

Copy code

```
printf("%%-6.1f = %-6.1f\n", 45.67);
printf("%%8d = %8d\n", 456);
printf("%%-6d = %-6d\n", 123);
printf("%%#10x = %#10x\n", 15);
printf("%%6.3f = %6.3f\n", 7.456);
printf("%%8.2f = %8.2f\n", 1234.56);
```

2. **Answers**:
    a. %-6.1f: 45.7
    b. %8d: 456
    c. %-6d: 123
    d. %#10x: 0xf
    e. %6.3f: 7.456
    f. %8.2f: 1234.56

Question 16

1. **Differentiate hexadecimal with and without prefix:**

c

Copy code

```
printf("%%5x = %5x\n", 255);
printf("%%#5x = %#5x\n", 255);
printf("%%8X = %8X\n", 127);
printf("%%#8X = %#8X\n", 127);
printf("%%6.2x = %6.2x\n", 45);
printf("%%#6.2x = %#6.2x\n", 45);
```

2. **Answers**:
    a. %5x: ff
    b. %#5x: 0xff
    c. %8X: 7F
    d. %#8X: 0X7F
    e. %6.2x: 2d
    f. %#6.2x: 0x2d

Question 17

1. **Experiment with zero-padded octal, hexadecimal, and floating-point:**

c

Copy code

```
printf("%%05o = %05o\n", 32);
printf("%%#06x = %#06x\n", 15);
printf("%%07.3f = %07.3f\n", 123.456);
printf("%%#08x = %#08x\n", 255);
printf("%%8.4o = %8.4o\n", 27);
```

2. **Answers**:

    a. %05o: 00040
    b. %#06x: 0x000f
    c. %07.3f: 123.456
    d. %#08x: 0x0000ff
    e. %8.4o: 0033

Question 18

1. **Examine width and precision in scientific notation:**

c

Copy code

```
printf("%%10.3e = %10.3e\n", 0.000567);
printf("%%-10.4e = %-10.4e\n", 0.0008765);
printf("%%7.1e = %7.1e\n", 3456.789);
printf("%%6.2e = %6.2e\n", 12345.67);
printf("%%12.4e = %12.4e\n", 98765.4321);
```

2. **Answers**:
    a. %10.3e: 5.670e-04
    b. %-10.4e: 8.765e-04
    c. %7.1e: 3.5e+03
    d. %6.2e: 1.2e+04
    e. %12.4e: 9.8765e+04

Question 19

1. **Format and align hexadecimal, octal, and integers:**

c

Copy code

```
printf("%%#-5o = %#-5o\n", 16);
printf("%%-6d = %-6d\n", 890);
printf("%%08d = %08d\n", 789);
printf("%%#6x = %#6x\n", 27);
printf("%%-8x = %-8x\n", 450);
```

2. **Answers**:
    a. %#-5o: 020
    b. %-6d: 890
    c. %08d: 00000789
    d. %#6x: 0x1b
    e. %-8x: 1c2

Question 20

1. **Explore format specifiers for multiple data types:**

c

Copy code

```
printf("%%.2f = %.2f\n", 45.6789);
printf("%%5.3g = %5.3g\n", 1234.5678);
printf("%%10.5f = %10.5f\n", 9.87654);
printf("%%#6o = %#6o\n", 28);
printf("%%#4x = %#4x\n", 31);
printf("%%.4e = %.4e\n", 123.45);
```

2. **Answers**:
    a. %.2f: 45.68
    b. %5.3g: 1230
    c. %10.5f: 9.87654
    d. %#6o: 034
    e. %#4x: 0x1f
    f. %.4e: 1.2345e+02