Answers are below the last question.

1) Trace by hand. How many times will the loop run?

int q = 5;

while (q > 0){

q -= q-1 % (q-2);

}

2) How many stars are output when the following code is executed?

for (int i = 0; i < 5; i++) {

for (int j = 0; j < 5; j++)

System.out.println("\*");

}

3) What is printed as a result of the following code segment?

for (int k = 0; k < 20; k+=2) {

if (k % 3 == 1)

System.out.print(k + " ");

}

4) What are the values of var1 and var2 after the following code segment is executed and the while loop finishes?

int var1 = 0;

int var2 = 2;

while ((var2 != 0) && ((var1 / var2) >= 0)) {

var1 = var1 + 1;

var2 = var2 - 1;

}

5) Write a method with a loop to compute and return how long it will take an investment to double in value at a given interest rate compounded annually.

6) Write a method to count and return the number of times it takes to roll all of the numbers between 1 and 6 on a die.

7) Write a loop to find the first n prime numbers.

1) Trace by hand. How many times will the loop run?

int q = 5;

while (q > 0){

q -= q-1 % (q-2);

}

2

2) How many stars are output when the following code is executed?

for (int i = 0; i < 5; i++) {

for (int j = 0; j < 5; j++)

System.out.println("\*");

}

25

3) What is printed as a result of the following code segment?

for (int k = 0; k < 20; k+=2) {

if (k % 3 == 1)

System.out.print(k + " ");

}

4 10 16

4) What are the values of var1 and var2 after the following code segment is executed and the while loop finishes?

int var1 = 0;

int var2 = 2;

while ((var2 != 0) && ((var1 / var2) >= 0)) {

var1 = var1 + 1;

var2 = var2 - 1;

}

var1 = 2, var2 = 0

5) Write a method with a loop to compute and return how long it will take an investment to double in value at a given interest rate compounded annually.

public static double timeToDoubleInvestment(double rate){

int t = 0;

double a = 1;

while (a < 2){

a\*=(1 + rate);

t++;

}

return t;

}

6) Write a method to count and return the number of times it takes to roll all of the numbers between 1 and 6 on a die.

public static int dieRollCounter() {

boolean one = false, two = false, three = false, four = false, five = false, six = false;

int count = 0;

while (!(one && two && three && four && five && six)){

int roll = (int)(Math.random()\*6 + 1);

if (roll == 1){

one = true;

} else if (roll == 2){

two = true;

} else if (roll == 3){

three = true;

} else if (roll == 4){

four = true;

} else if (roll == 5){

five = true;

} else if (roll == 6){

six = true;

}

count++;

}

return count;

}

7) Write a method to print the first n prime numbers.

public static void printPrimeNums(int num){

if (num > 0){

System.out.print(2 + " ");

num--;

}

int possiblePrime = 3;

while (num > 0){

boolean prime = true;

for (int div = (int)(Math.round(Math.sqrt(possiblePrime))); div > 1; div--){

if (possiblePrime % div == 0){

prime = false;

}

}

if (prime){

System.out.print(possiblePrime + " " );

num--;

}

possiblePrime += 2;

}

}