**Chapter 5 – Answers**

**5.7 Values of x:**

a) 7.5  
b) 7.0  
c) 0.0  
d) 0.0  
e) 6.4  
f) -6.0  
g) -9.0

**5.12 Random integers for variable n:**

a) n = 1 + rand.nextInt(2);  
b) n = 1 + rand.nextInt(100);  
c) n = rand.nextInt(10);  
d) n = 1000 + rand.nextInt(113);  
e) n = -1 + rand.nextInt(3);  
f) n = -3 + rand.nextInt(15);

**5.13 Random numbers from sets:**

a) n = 2 + 2 \* rand.nextInt(5); // 2, 4, 6, 8, 10  
b) n = 3 + 2 \* rand.nextInt(5); // 3, 5, 7, 9, 11  
c) n = 6 + 4 \* rand.nextInt(5); // 6, 10, 14, 18, 22

**5.14 Exponentiation:**

int integerPower(int base, int exponent) {

int result = 1;

for (int i = 1; i <= exponent; i++) {

result \*= base;

}

return result;

}

**5.15 Hypotenuse Calculation:**

double hypotenuse(double side1, double side2) {

return Math.sqrt(Math.pow(side1, 2) + Math.pow(side2, 2));

}

**5.16 Multiples:**

boolean isMultiple(int a, int b) {

return b % a == 0;

}

**5.17 Even or Odd:**

boolean isEven(int number) {

return number % 2 == 0;

}

**5.18 Square of Asterisks:**

void squareOfAsterisks(int side) {

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

for (int j = 0; j < side; j++) {

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

}

System.out.println();

}

}

**5.19 Square with Character:**

void squareOfAsterisks(int side, char fillCharacter) {

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

for (int j = 0; j < side; j++) {

System.out.print(fillCharacter);

}

System.out.println();

}

}

**5.20 Circle Area:**

double circleArea(double radius) {

return Math.PI \* radius \* radius;

}

**5.21 Separating Digits:**

a) int quotient = a / b;  
b) int remainder = a % b;  
c)

void displayDigits(int number) {

String numStr = Integer.toString(number);

for (int i = 0; i < numStr.length(); i++) {

System.out.print(numStr.charAt(i) + " ");

}

}

**5.22 Temperature Conversions:**

a)

double celsius(double f) {

return 5.0 / 9.0 \* (f - 32);

}

b)

double fahrenheit(double c) {

return 9.0 / 5.0 \* c + 32;

}

**5.23 Minimum of Three:**

double minimum3(double x, double y, double z) {

return Math.min(x, Math.min(y, z));

}

**5.24 Perfect Numbers:**

boolean isPerfect(int num) {

int sum = 0;

for (int i = 1; i < num; i++) {

if (num % i == 0) sum += i;

}

return sum == num;

}

**5.25 Prime Numbers:**

a)

boolean isPrime(int n) {

if (n < 2) return false;

for (int i = 2; i <= Math.sqrt(n); i++) {

if (n % i == 0) return false;

}

return true;

}

**5.26 Reverse Digits:**

int reverse(int number) {

int reversed = 0;

while (number != 0) {

reversed = reversed \* 10 + number % 10;

number /= 10;

}

return reversed;

}

**5.27 GCD:**

int gcd(int a, int b) {

while (b != 0) {

int temp = b;

b = a % b;

a = temp;

}

return a;

}

**5.28 Quality Points:**

int qualityPoints(int avg) {

if (avg >= 90) return 4;

if (avg >= 80) return 3;

if (avg >= 70) return 2;

if (avg >= 60) return 1;

return 0;

}

**5.29 Coin Tossing:**

enum Coin { HEADS, TAILS; }

Coin flip() {

Random rand = new Random();

return rand.nextBoolean() ? Coin.HEADS : Coin.TAILS;

}

**5.30 Guess the Number:**

int secret = 1 + rand.nextInt(1000);

while (true) {

int guess = input.nextInt();

if (guess < secret) System.out.println("Too low. Try again.");

else if (guess > secret) System.out.println("Too high. Try again.");

else {

System.out.println("Congratulations. You guessed the number!");

break;

}

}

**5.31 Guess with Feedback:**

int count = 0;

while (true) {

count++;

int guess = input.nextInt();

if (guess < secret) System.out.println("Too low. Try again.");

else if (guess > secret) System.out.println("Too high. Try again.");

else {

System.out.println("Congratulations. You guessed the number!");

break;

}

}

if (count < 10) System.out.println("Either you know the secret or you got lucky!");

else if (count == 10) System.out.println("Aha! You know the secret!");

else System.out.println("You should be able to do better!");

**5.32 Distance Between Points:**

double distance(double x1, double y1, double x2, double y2) {

return Math.sqrt(Math.pow(x2 - x1, 2) + Math.pow(y2 - y1, 2));

}

**5.33 Craps with Wager:**

* Initialize bankBalance = 1000
* Prompt for wager: if wager > bankBalance, ask again
* If win: bankBalance += wager
* If lose: bankBalance -= wager
* If bankBalance == 0, print “Sorry. You busted!”
* Add chatter with random responses using a method like displayChatter()

**5.34 Binary, Octal, Hex Table:**

for (int i = 1; i <= 256; i++) {

System.out.printf("%d %s %o %x%n", i, Integer.toBinaryString(i), i, i);

}

**5.35–5.39 Computer-Assisted Instruction:**

**5.35**: Prompt multiplication question with random one-digit numbers, repeat if wrong.  
**5.36**: Use random response from a set for correct/incorrect answers.  
**5.37**: Track 10 answers, show % score