**1.a**

1. Algorithm arrayMax(A,n):
2. Input: Taulukko A sisältää n kokonaislukua.
3. Output: Taulukon maksimialkio.
4. currentMax 🡨 A [0]
5. for i 🡨 1 to n-1 do
6. if currentMax<A[i] then
7. currentMax 🡨 A[i]
8. return currentMax

**b**

1. Algorithm arrayMin(A,n):
2. Input: Taulukko A sisältää n kokonaislukua.
3. Output: Taulukon minimialkio.
4. currentMin 🡨 A[0]
5. for i 🡨 1 to n-1 do
6. if currentMin < A[i] then
7. currentMin 🡨 A[i]
8. return currentMin

**2.a**

1. package  tira18\_01;
2. import  java.io.\*;
3. import  java.util.Scanner;
4. /\*\*
5. \*
6. \* @author tuomo
7. \*/
8. public  class  Tira18\_01\_2a  {
9. /\*\*
10. \* @param args the command line arguments
11. \*/
12. public  static  void  main(String[]  args)  {
13. Scanner  scan  =  new  Scanner(System.in);
14. if  (args.length  ==  1)  {
15. try  {  //Yrittää löytää annettu tekstitiedosto
16. File  txtfile  =  new  File(args[0]);
17. Scanner  lukija  =  new  Scanner(txtfile);
18. int  currentMax  =  0;
19. while  (lukija.hasNextInt())  {  //Lukija lukee seuraavan luvun                         int  num  =  lukija.nextInt();
20. if  (num  >  currentMax)  {  //Jos luku on suurempi kuin suurin luku
21. currentMax  =  num;      //siitä tulee suurin luku
22. }
23. }
24. System.out.println(currentMax);  //Lopuksi tulostetaan suurin luku
26. }
27. catch  (FileNotFoundException  ex)  {  //Tiedostoa ei löydetty
28. tulostetaan viesti ja lopetetaan ohjelma
29. System.out.println("Invalid command-line argument!");
30. }
31. System.out.println("Bye, see you soon.");
32. }
33. else  {  //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
34. System.out.println("Invalid command-line argument!");
35. System.out.println("Bye, see you soon.");
36. }
37. }
38. }

**b**

1. package tira18\_01;
2. import java.io.\*;
3. import java.util.Scanner;
4. /\*\* \* \* @author tuomo \*/
5. public class Tira18\_01\_2b {
6. /\*\*     \* @param args the command line arguments     \*/
7. public static void main(String[] args) {
8. Scanner scan = new Scanner(System.in);
9. if (args.length == 1) {
10. try { //Yrittää löytää annettu tekstitiedosto
11. File txtfile = new File(args[0]);
12. Scanner lukija = new Scanner(txtfile);
13. int currentMin = Integer.MAX\_VALUE;
14. while (lukija.hasNextInt()) { //Lukija lukee seuraavan luvun
15. int num = lukija.nextInt();
16. if (num < currentMin) { //Jos luku on pienempi kuin suurin luku, siitä tulee pienin luku
17. currentMin = num;
18. }
19. }
20. System.out.println(currentMin); //Lopuksi tulostetaan pienin luku
21. } catch (FileNotFoundException ex) { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
22. System.out.println("Invalid command-line argument!");
23. }
24. System.out.println("Bye, see you soon.");
25. } else { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
26. System.out.println("Invalid command-line argument!");
27. System.out.println("Bye, see you soon.");
28. }
29. }
30. }

**3a**

1. Algorithim aritmeettinen(A, n):
2. Input: Taulukko A sisältää n kokonaislukua.
3. Output: Taulukon aritmeettinen keskiarvo.
4. sum🡨A[0]
5. counter🡨A[0]
6. for i(1 to n - 1 do
7. sum 🡨 sum + n
8. counter++
9. average 🡨 sum / counter
10. return average

**b**

1. Algorithim harmoninen(A, n):
2. Input: Taulukko A sisältää n kokonaislukua.
3. Output: Taulukon harmoninen keskiarvo.
4. sum🡨A[0]
5. counter🡨A[0]
6. for i(1 to n - 1 do
7. sum 🡨 sum + 1 / n
8. counter++
9. average 🡨 counter / sum
10. return average

**c**

**4a**

1. package tira18\_01;
2. import java.io.\*;
3. import java.util.Scanner;
4. /\*\* \* \* @author tuomo \*/
5. public class Tira18\_01\_4a {
6. /\*\*     \* @param args the command line arguments     \*/
7. public static void main(String[] args) {
8. Scanner scan = new Scanner(System.in);
9. if (args.length == 1) {
10. try { //Yrittää löytää annettu tekstitiedosto
11. File txtfile = new File(args[0]);
12. Scanner lukija = new Scanner(txtfile);
13. int counter = 0;
14. int sum = 0;
15. while (lukija.hasNextInt()) { //Lukija lukee seuraavan luvun
16. int num = lukija.nextInt();
17. sum = sum + num;
18. counter++;
19. }
20. double average = (sum \* 1.0) / (counter \* 1.0);
21. System.out.println(average); //Lopuksi tulostetaan keskiarvo
22. } catch (FileNotFoundException ex) { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
23. System.out.println("Invalid command-line argument!");
24. }
25. System.out.println("Bye, see you soon.");
26. } else { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
27. System.out.println("Invalid command-line argument!");
28. System.out.println("Bye, see you soon.");
29. }
30. }
31. }

**b**

1. package tira18\_01;
2. import java.io.File;
3. import java.io.FileNotFoundException;
4. import java.util.Scanner;
5. /\*\* \* \* @author tuomo \*/
6. public class Tira18\_01\_4b {
7. /\*\*     \* @param args the command line arguments     \*/
8. public static void main(String[] args) {
9. Scanner scan = new Scanner(System.in);
10. if (args.length == 1) {
11. try { //Yrittää löytää annettu tekstitiedosto
12. File txtfile = new File(args[0]);
13. Scanner lukija = new Scanner(txtfile);
14. int counter = 0;
15. double sum = 0;
16. while (lukija.hasNextInt()) { //Lukija lukee seuraavan luvun
17. double num = lukija.nextInt();
18. sum = sum + (1 / num);
19. counter++;
20. }
21. double average = (counter \* 1.0) / sum;
22. System.out.println(average); //Lopuksi tulostetaan keskiarvo
23. } catch (FileNotFoundException ex) { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
24. System.out.println("Invalid command-line argument!");
25. }
26. System.out.println("Bye, see you soon.");
27. } else { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
28. System.out.println("Invalid command-line argument!");
29. System.out.println("Bye, see you soon.");
30. }
31. }
32. }

**c**

1. package tira18\_01;
2. import java.io.File;
3. import java.io.FileNotFoundException;
4. import java.util.Scanner;
5. /\*\* \* \* @author tuomo \*/
6. public class Tira18\_01\_4c {
7. /\*\*     \* @param args the command line arguments     \*/
8. public static void main(String[] args) {
9. Scanner scan = new Scanner(System.in);
10. if (args.length == 1) {
11. try { //Yrittää löytää annettu tekstitiedosto
12. File txtfile = new File(args[0]);
13. Scanner lukija = new Scanner(txtfile);
14. int counter = 0;
15. int sum = 0;
16. while (lukija.hasNextInt()) { //Lukija lukee seuraavan luvun
17. int num = lukija.nextInt();
18. sum = sum + num;
19. counter++;
20. }
21. double average = Math.pow(sum, 1.0 / counter); //(sum\*1.0)/(counter\*1.0);
22. System.out.println(average); //Lopuksi tulostetaan keskiarvo
23. } catch (FileNotFoundException ex) { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
24. System.out.println("Invalid command-line argument!");
25. }
26. System.out.println("Bye, see you soon.");
27. } else { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
28. System.out.println("Invalid command-line argument!");
29. System.out.println("Bye, see you soon.");
30. }
31. }
32. }

**5**

**6**

Algorithm Median(A,n)

Input: Taulukko A sisältää n kokonaislukua

Output: Taulukon mediaani

Return median:

**7**

1. package tira18\_01;
2. import java.io.File;
3. import java.io.FileNotFoundException;
4. import java.util.Arrays;
5. import java.util.Scanner;
6. /\*\* \* \* @author tuomo \*/
7. public class Tira18\_01\_7a {
8. /\*\*     \* @param args the command line arguments     \* @throws java.lang.Exception     \*/
9. public static void main(String[] args) {
10. if (args.length == 1) {
11. try { //Yrittää löytää annettu tekstitiedosto
12. File txtfile = new File(args[0]);
13. int Median = Median(txtfile);
14. System.out.println(Median);
15. } catch (FileNotFoundException ex) { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
16. System.out.println("Invalid command-line argument!");
17. }
18. System.out.println("Bye, see you soon.");
19. } else { //Tiedostoa ei löydetty tulostetaan viesti ja lopetetaan ohjelma
20. System.out.println("Invalid command-line argument!");
21. System.out.println("Bye, see you soon.");
22. }
23. }
24. public static int Median(File txtfile) throws FileNotFoundException {
25. Scanner scan = new Scanner(System.in);
26. Scanner lukija = new Scanner(txtfile);
27. int numberOfElements = lukija.nextInt();
28. int[] numbers = new int[numberOfElements];
29. for (int i = 0; i < numberOfElements; ++i) {
30. numbers[i] = lukija.nextInt();
31. }
32. Arrays.sort(numbers);
33. int medianIndex = ((numberOfElements % 2 == 0 ? numberOfElements : numberOfElements + 1) >> 1) - 1;
34. int fin = numbers[medianIndex];
35. return fin;
36. }
37. }

**8**