Problem 11.1

```
public class problem11_1 {
	private String Color;
	private boolean isfilled;

	public String getColor() {
		return Color;
	}
	public void setFill(boolean fill) {
		isfilled = fill;
	}
	public boolean isFilled() {
		return isfilled;
	}

}

import java.util.Scanner;

public class Triangle extends problem11_1{

	private double side1;
	private double side2;
	private double side3;

	public static void main(String[] args) {
		double s1,s2,s3;
		String color;
		boolean fill;

		Scanner input = new Scanner(System.in);
		System.out.println("Enter color: ");
		color = input.nextLine();

		System.out.println("Enter sides: ");
		s1 = input.nextDouble();
		s2 = input.nextDouble();
		s3 = input.nextDouble();
```

```
        System.out.println("Fill triangle or not: ");
        fill = input.nextBoolean();

        Triangle tt = new Triangle (s1,s2,s3);
        tt.putcolor(color);
        tt.setFill(fill);

        System.out.println("Sides of "+ tt.toString());
        System.out.println("Area of Triangle "+tt.getarea());
        System.out.println("Perimeter of Triangle " +tt.getperimeter());
        System.out.println("color of the Triangle is "+ tt.getColor());
        System.out.println("is Triangle filles " +tt.isFilled());

}
public Triangle (double s1, double s2, double s3) {
        side1 = s1;
        side2 = s2;
        side3 = s3;
}
public double getside1() {
        return side1;
}
public double getside2() {
        return side2;
}
public double getside3() {
        return side3;
}
public double getarea() {
        double s = (side1+side2+side3)/2;
        double area = s*(s-side1)*(s-side2)*(s-side3);
        return area;
}
public double getperimeter() {
        return (side1+side2+side3);
}
public String toString() {
        return "Triangle: side1="+side1+" side2= "+side2+ "side3=" +side3;
```

```
        }

}
```

Problem 11.3

```
        class Account {
        private int number;
        private double balance;
        private double annualInterestRate;

        private java.util.Date dateCreated = new java.util.Date();

        public Account() {
                java.util.Date dateCreated = new java.util.Date();
        }
        public Account (int id, double bal, double AIRate) {
                number = id;
                balance = bal;
                annualInterestRate = AIRate;
        }
        public void setdate(java.util.Date dateCreated) {
                this.dateCreated =dateCreated;
        }
        public java.util.Date getdate() {
                return dateCreated;
        }
        public int getID() {
                return number;
        }
        public double getBalance() {
                return balance;
        }
        public double deposit (double deposit) {
                balance += deposit;
                return(deposit);
        }
        public double withdraw (double withdraw) {
```

```
                balance -= withdraw;
                return(withdraw);
        }
        public double getAnnualInterestRate() {
                return annualInterestRate;
        }
        public void setAnnualInterestRate(double annualInterestrate) {
                this.annualInterestRate = annualInterestRate;
        }
        public double getMonthlyInterestRate() {
                double monthlyInterest = (annualInterestRate/120) * balance;
                return monthlyInterest;
        }
}


class CheckingsAccount extends Account {
        int overdraft = 500;
        public String toString() {
                return "Checkings Account " + "overdraft limit " + overdraft;
        }
}

class SavingsAccount extends Account {
                int overdraft = 0;
                public String toString() {
                        return "Savings Account " + "overdraft limit " + overdraft;
                }
}


public class TestA {

        public static void main(String[] args) {
                Account account1 = new Account(1001,10000, 2.9);
                Account account2 = new SavingsAccount();
                Account account3 = new CheckingsAccount();

                account1.deposit(5000);
```

```
                account1.withdraw(4500);

                System.out.println("\nAccount ID: " + account1.getID());
                System.out.println("Balance: " + account1.getBalance());
                System.out.println("Monthly interest rate: " +account1.getMonthlyInterestRate());
                System.out.println("Date created: " + account1.getdate());
                System.out.println("toString: " + account1.toString());
                System.out.println("toString: " + account2.toString());
                System.out.println("toString: " + account3.toString());

        }

}
```

Problem 11.5

```
import java.util.ArrayList;

class Course {
        private String CName;
        private ArrayList students;
        public Course (String CN) {
                CName = CN;
                students = new ArrayList();
        }
        public void addStudent (String stu) {
                students.add(stu);
        }
        public ArrayList getStudents() {
                return students;
        }
        public int getNumberOfStudents() {
                return students.size();
        }
        public String getCourseName() {
                return CName;
        }
        public void dropStudent (String stu) {
                students.remove(stu);
        }
```

```
	public String toString() {
		String s = "";
		s = CName + "(" + students.size() + "students)\n";
		for(int i = 0; i < students.size(); i++) {
			s += "(" + (i+1) + ") " + students.get(i) +
					"\n";
		}
		return s;
	}
	public void clear() {
		students.clear();
	}
}

import java.util.ArrayList;

public class TestC {

	public static void main(String[] args) {
		Course c1 = new Course ("Software Devepolment1");
		Course c2 = new Course ("Software Devepolment2");

		c1.addStudent("Theodore Higgins");
		c1.addStudent("Eileen Hayes");
		c1.addStudent("Bertha King");
		c1.addStudent("Julia Bell");

		c2.addStudent("Lyle Brady");
		c2.addStudent("Bennie Franklin");
		c2.addStudent("Dustin Colon");

		System.out.println("Number of students in course1: " +
c1.getNumberOfStudents());
		ArrayList a1 = c1.getStudents();
		for(int i = 0; i <c1.getNumberOfStudents(); i++)
			System.out.print(a1.get(i)+ ", ");
		System.out.println();
		System.out.print("Number of Students in course2:" +
c2.getNumberOfStudents());
```

```
            System.out.println();
            ArrayList a2 = c2.getStudents();
            for (int i=0; i < c2.getNumberOfStudents(); i++)
                  System.out.print(a2.get(i) + ", ");
            System.out.println();
            c1.dropStudent("Julia Bell");
            System.out.println("\nNew list of course 1 after drop a student Julia Bell: " + c1);
            c2.clear();
            System.out.println("\nNew Course2 list after using clear() method: " + c2);
      }

}
```

Problem 11.11

```
import java.util.ArrayList;
import java.util.Scanner;

public class problem11_11 {

      public static void main(String[] args) {
            ArrayList<Integer> list = new ArrayList<Integer>();
            Scanner input = new Scanner (System.in);

            System.out.print("Enter 5 numbers: ");
            int i=0;
            while (i<5) {
                  list.add(input.nextInt());
                  i++;
            }
            sort(list);
            System.out.print("List after sorting: ");
            for(int j=0; j<list.size(); j++)
                  System.out.print(list.get(j)+" ");
      }
      public static void sort (ArrayList<Integer> list) {
            int temp;
            for(int i=0; i<list.size(); i++) {
```

```
                    for(int j=i+1; j<list.size()-1; j++) {
                        if(list.get(i)>list.get(j)) {
                            temp=list.get(i);
                            list.set(i, list.get(j));
                            list.set(j, temp);
                        }
                    }
                }
            }

}
```

Problem 11.13

```
import java.util.ArrayList;
import java.util.Scanner;

public class problem11_13 {
    public static void main(String[] args) {
        ArrayList list = new ArrayList();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter ten integers: ");

        for(int i=0; i<10; i++) {
            list.add(scanner.nextInt());
        }
        removeDuplicate(list);
        System.out.print("The distinct integers are ");
        for (int i=0; i<list.size();i++) {
            System.out.print(list.get(i)+ " ");
        }
    }
    public static void removeDuplicate(ArrayList list) {
        for (int i=0; i<list.size()-1;i++) {
            for(int j=i+1; j<list.size(); j++) {
                if(list.get(i) == list.get(j)) {
                    list.remove(j);
                }
            }
```

```
		}
	}

}
```

Problem 11.17

```
import java.util.ArrayList;
import java.util.Scanner;

public class problem11_17 {
	public static void main(String[] args) {
		Scanner scanner = new Scanner(System.in);
		System.out.print("Enter an integre m: ");

		int m = scanner.nextInt();
		ArrayList integerFactors = new ArrayList();

		computeIntegerFactors(m, integerFactors);
		getOddNumberedFactors(integerFactors);

		int smallestNumber = 1;

		for(int i=0; i<integerFactors.size();i++) {
			smallestNumber = smallestNumber * integerFactors.get(i);
		}
		System.out.println("The smallest number n for m *n to be a perfect square is "+
smallestNumber);
		System.out.println("m * n is " +(m*smallestNumber));
	}
	private static void computeIntegerFactors(int m, ArrayList integerFactors) {
			int factor = 2;
			while (factor <= m) {
				if (m % factor == 0)
				integerFactors.add(factor);
				m /= factor;
			}
			{
```

```
                        factor++;
                    }
                }

        public static void getOddNumberedFactors(ArrayList list) {
                for (int i=0; i<list.size()-1; i++) {
                    for(int j=i+1; j<list.size(); j++) {
                        if(list.get(i) == list.get(j)) {
                            list.remove(j);
                            list.remove(i);
                        }
                    }
                }

        }


}
```