

IIP  
Test Unit 5 - Possible solution  
Year 2013-2014

Name:

1. Implement a datatype class **Computer** that represents computer features. You must develop:

- a) Attributes for processor type (String), number of processors (int), bits of the architecture (int), clock frequency in GHz (real), and RAM memory capacity in gigabytes (real).
- b) A constructor that receives processor type, bits of the architecture, and clock frequency; default number of processors is 1 and default memory capacity is 1Gb.
- c) A constructor that receives the data necessary for initing all the attributes.
- d) The **get** and **set** methods for each attribute.
- e) An **equals** method that overrides the functionality of the method of the **Object** class; you must check the values of all the attributes.
- f) A **toString** method that returns the string in format: "Computer TYPE of BITS bits with NUM processors at CLOCK GHz - RAM memory of CAPACITY Gb"
- g) A method that returns the consumption in watts of the computer, calculated as clock frequency times number of processors.
- h) A method that returns if the computer has enough space to store a program in memory, given the size of the program in Mb as parameter.

**Note:** all the constructors and **set** methods must check that the numeric attributes are positive; in other case, constructors must assign reasonable values to the attributes, and **set** methods will not modify the corresponding attributes.

```
public class Computer {
    private String procType;
    private int numProc, numBits;
    private double freq, ram;

    public Computer(String t, int b, double f) {
        procType=new String(t);
        if (b>0) numBits=b; else numBits=32;
        if (f>0) freq=f; else freq=1.0;
        numProc=1;
        ram=1.0;
    }

    public Computer(String t, int np, int b, double f, double r) {
        procType=new String(t);
        if (np>0) numProc=np; else numProc=1;
        if (b>0) numBits=b; else numBits=32;
        if (f>0) freq=f; else freq=1.0;
        if (r>0) ram=r; else ram=1.0;
    }

    public String getProcType() { return procType; }
    public int getNumProc() { return numProc; }
    public int getNumBits() { return numBits; }
    public double getFreq() { return freq; }
    public double getRam() { return ram; }

    public void setProcType(String t) { procType=new String(t); }
    public void setNumProc(int np) { if (np>0) numProc=np; }
    public void setNumBits(int b) { if (b>0) numBits=b; }
    public void setFreq(double f) { if (f>0) freq=f; }
    public void setRam(double r) { if (r>0) ram=r; }

    public boolean equals(Object o) {
        return o instanceof Computer &&
            procType.equals(((Computer) o).procType) &&
            numProc==((Computer) o).numProc &&
            numBits==((Computer) o).numBits &&
            freq==((Computer) o).freq &&
            ram==((Computer) o).ram;
    }
}
```

```

public String toString() {
    return "Computer "+procType+" of "+numBits+" bits with "+numProc+" processors at "+freq+
        " GHz - RAM memory of "+ram+"Gb";
}

public double consumption() { return freq*numProc; }

public boolean enoughSpace(double size) { return (ram*1024>size); }

}

```

2. Implement a program class that has a `main` method that asks for the data of two computers (only processor type, bits, and clock frequency) and calls another `static` method (in the same class) that receives as parameters the two computers, and shows on the screen the ratio (division) between clock frequencies of the two computers.

```

import java.util.*;

public class TestComputer {

    public static void main(String [] args) {
        Scanner kbd=new Scanner(System.in).useLocale(Locale.US);
        Computer c1, c2;
        String t;
        int bits;
        double freq;

        System.out.print("Computer 1 - Type: "); t=kbd.nextLine();
        System.out.print("Computer 1 - Bits: "); bits=kbd.nextInt();
        System.out.print("Computer 1 - Frequency: "); freq=kbd.nextDouble();
        c1=new Computer(t,bits,freq);

        kbd.nextLine();
        System.out.print("Computer 2 - Type: "); t=kbd.nextLine();
        System.out.print("Computer 2 - Bits: "); bits=kbd.nextInt();
        System.out.print("Computer 2 - Frequency: "); freq=kbd.nextDouble();
        c2=new Computer(t,bits,freq);

        ratio(c1,c2);
    }

    public static void ratio(Computer c1, Computer c2) {
        System.out.println("Clock ratio between computers 1 and 2: "+(c1.getFreq()/c2.getFreq()));
    }
}

```