CS 1401, Exam #3, TR version

4-2

$\frac{50}{50} = \frac{100}{100}$

Date: Thursday, November 7, 2013
Name (please type legibly, ideally in block letters): [redacted]

On November 7, 1867, the famous scientist Marie Curie was born. Dr. Curie, a French-Polish physicist and chemist, is famous for her pioneering research on radioactivity. She was the first woman to win a Nobel Prize, the only woman to win in two fields, and the only person to win in multiple sciences.

1a. Write a method named *energy* that, given the power *p* of a radioactive source (in Watts W) and the time *t* (in seconds) that it operates, returns the amount of energy produce by this source. (Hint: power is energy per unit time, so energy is power times time).

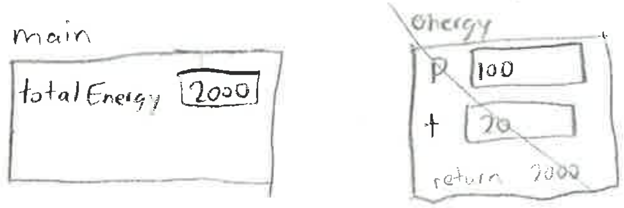
1b. Call (invoke) your method *energy* in the *main* method to compute the total energy produced by the 100 Watt source during 20 seconds. You do not need to write the entire *main* method, just the part that assigns values to the corresponding variables *power* and *time* and calls your method.

$\frac{10}{10}$ 1c. Trace, step by step, how the computer will perform the needed computations, and check that the result is indeed correct.

a)
```
public static double energy(double p, double t)
{
  return p* t;
}
```

b)
```
double totalEnergy = energy (100, 20);
```

c)



2a. Define a class *Nobel* whose objects are different Nobel prize winners. The description of each winner should include his/her name, birth year, and the year in which he/she was awarded the prize. Your class should contain a constructor method, get- and set-methods, and a method for computing the age at which the scientist received the Nobel Prize as the difference between the two years.

10/10

2b. Use your class in the *main* method to define a new object *curie* of type *Nobel.* Marie Curie was born in 1867, she received her first Nobel Prize in 1903. Compute and print her age at the time of receiving the Nobel Prize; then, replace 1903 with the date of her second Nobel Prize (1911) and compute and print her age at the time of the second prize. on back

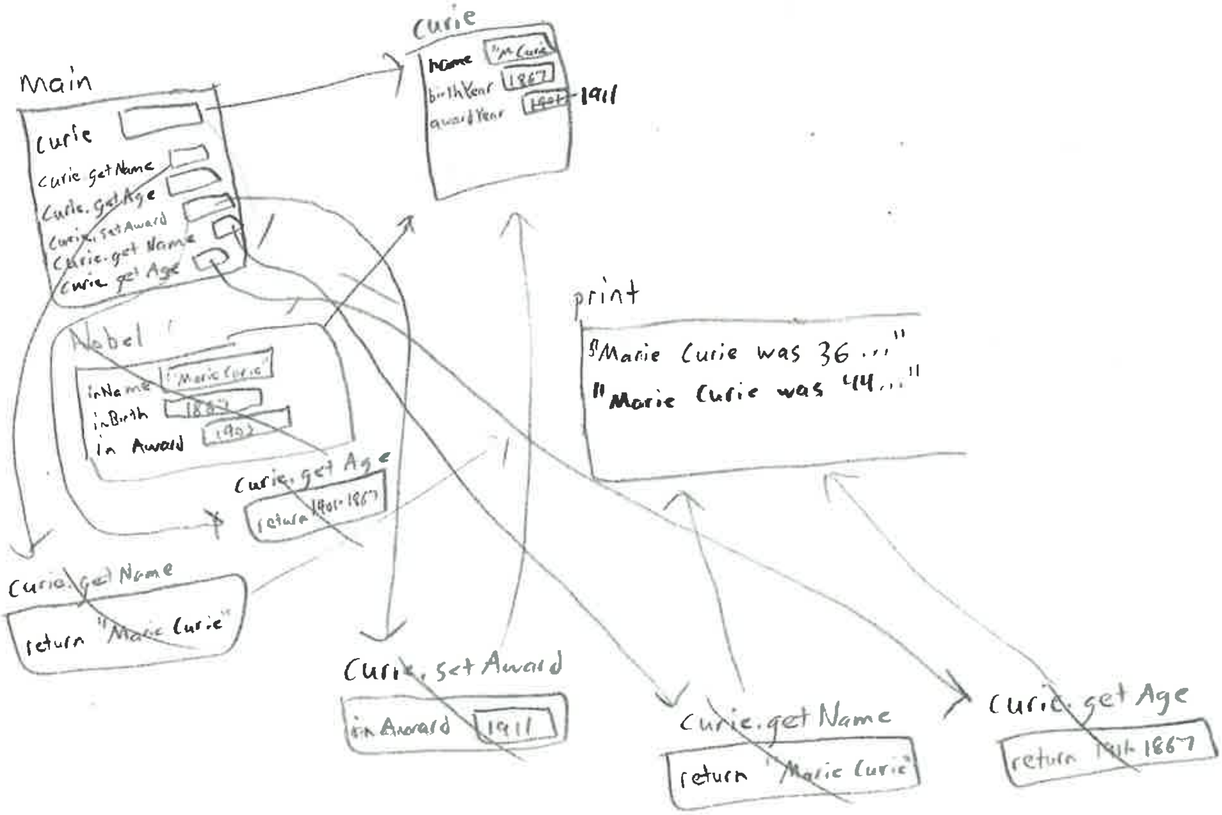
2c. Trace your program step-by-step. on back

a)
```
public class Nobel
{ // declare variables
   private String name;
   private int birthYear, awardYear;

  // constructor for no inputs
  public Nobel()
  {
    name= "No name";
    birthYear= 0;
    awardYear =0;
  }
 // constructor for only String input
 public Nobel (String inName)
 {
  name= inName;
  birthYear= 0;
  awardYear= 0;
 }
 //constructor for only one int input (assume award year)
 public Nobel ( int inAward)
 {
  name= "No Name";
  birthYear =0;
  awardYear= inAward;
 }
 //constructor for 2 int inputs
 public Nobel(int inBirth, int inAward)
 {
  name= "No Name";
  birthYear= inBirth;
  awardYear = in Award;
 }

 // constructor for string and one int (assume award)
 public Nobel (String inName, int inAward)
 {
  name= inName;
  birthYear= 0;
  awardYear= inAward;
 }
 // constructor for all 3 inputs
 public Nobel(String inName, int inBirth, int inAward)
 {
  name = in Name;
  birthYear= inBirth;
  awardYear= inAward;
 }
     //get methods
     public String getName()
     { return name; }
     public int getBirth()
     { return birthYear; }
     public int getAward()
     { return awardYear; }
     //set Methods
     public void setName(String inName)
     { name = inName; }
     public void setBirth(int inBirth)
     { birthYear= inBirth; }
     public void setAward(int inAward)
     { awardYear= in Award; }
     //get age when received award
     public int getAge();
     { return (awardYear - birthYear); }
}
```

b) main

```
{
  Nobel curie= new Nobel("Marie Curie", 1867, 1903);
  System.out.println(curie.getName()+" was "+curie.getAge()+" when she won.");
  curie.setAward(1911);
  System.out.println(curie.getName()+" was "+ curie.getAge()+" when she won again.");
}
```

c)



```
"Marie Curie was 36..."
"Marie Curie was "44..."
```

(sorry its messy)

int

3a. Write a piece of code that, given an array *birthYear* of birth years and the current year *curYear*, computes and prints the ages of these students. Assume the arrays have been declared, initialized, and that they have the same length.

10/10

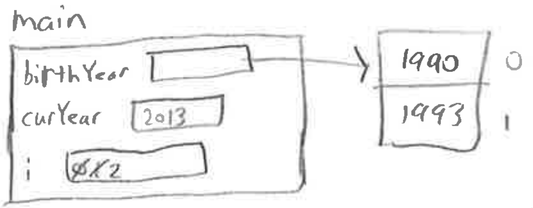
3b. To check the correctness of the code you wrote in Part 3a, write a piece of code that defines a new array *birthYear* with elements 1990 and 1993 and a new value *curYear* equal to 2013.

3c. Trace step-by-step how the piece of code you wrote in Part 3a will compute the corresponding ages.

a)
```
for(int i=0; i< birthYear.length; i++)
{
  System, out. println ("Student "+(i+1)+" is "+(curYear-birthYear[i])+" years old.";
}
```

b)
```
int[] birthYear = {1990, 1993};
int   curYear = 2013;
```

c)



```
SoPln ("Student 1 is 23 years old.");
SoPln ("Student 2 is 20 years old.");
```

4a. Write a method that, given an array *a* of ages of different students and the array *n* of their names, finds the name of the youngest student. Assume the arrays have been declared, initialized, and have the same non-zero length.

4b. To check the correctness of your method, write a piece of code that defines new arrays *age* consisting of 3 elements 23, 20, and 25, and *names* with elements K, L, and V.

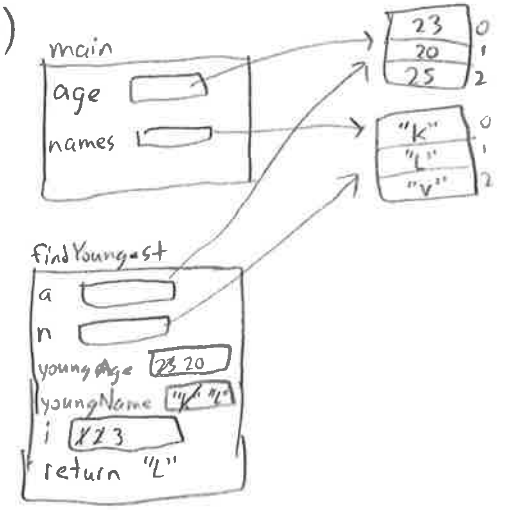
10/10

4c. Trace step-by-step how the piece of code you wrote in Part 4a will find the name of the youngest student. *Hint:* L is the youngest, he is only 20 years old.

a)
```
public static String findYoungest(int[] a, String[] n)
{
    int YoungAge = a[0];
    String young Name = n[0];
    for (int i=1; i< a.length; i++)
    {
        if(a[i] < youngAge)
        {
            young Age = a[i];
            YoungName = n[i];
        }
    }
    return YoungName;
}
```

b)
```
int[] age = {23, 20, 25};
String[] names = {"K", "L", "V"};
```

c)



5a. Describe what is white-box and black-box testing. 10

5b. Describe the main rules for testing. 10

a) white box testing is when you see the code, while black box you only see the inputs and outputs.

b) check the extreme values and some intermediate values of loops and logic tests if white box testing, test every branch of a program