ISTE-120 - Computational Problem Solving for the Information Domain I Homework Assignment 11 (HW11) - Inheritance

NOTE: Use the Scanner class for all input and printf() to format output.

Problem

Create a Student class with:

- string attributes name and id and
- integer numCredits
- Include mutators and accessors for each attribute and
- two constructors:
 - parameterized constructor uses mutators to set the attribute values
 - default constructor calls the parameterized constructor with default values

Class Student will need two public methods. The first method is calcTuition and simply returns 0.0. The second is toString that uses the accessor methods to get the attribute values, and returns student info as a single formatted string for output as shown below.

Create a second class named Undergrad that extends Student. Add:

- an integer attribute for yearLevel (1-4) plus
 - o the mutator and accessor methods for this new attribute

While a default constructor is not required, a full parameterized constructor with four arguments is required with:

- a call to the parent class parameterized constructor and
- a call to the mutator for yearLevel

Add two public methods:

- calcTuition that returns the product of numCredits and the undergraduate cost per credit hour (\$517 should be implemented as a constant)
- toString that returns the parent Student info as well as the yearLevel in a single formatted string shown below

Create a third class named Grad that extends the Student class. Add:

- a String attribute for researchArea
- o the mutator and accessor methods for this attribute Include a default constructor that uses the mutator for researchArea. A parameterized constructor is not needed for this class. Add two public methods:
 - calcTuition that returns the product of numCredits and the graduate cost per credit hour (\$713 should be a constant)
 - toString returns the parent student info as well as the research area in a single formatted string for output as shown in the sample below

Finally, create a StudentTest class with a main method. In the main method, prompt the user to enter a student type using a menu system as shown in the sample run below.

For Undergrad students, create the object using the full-parameterized constructor object. For Grad students, create the object using the default constructor and the mutator methods. Store both Student types in an ArrayList of Students named enrollment.

Once the user is done entering students, run through the arraylist using a for/each loop. Each time, print out the type of student, the results of a single call to a toString method, and the results of a call to the calcTuition method. Be sure the output is formatted as shown in the example. Use System.out.printf or String.format to align the output.

Sample Execution

```
Command Prompt
dkpvcs> java StudentTest
What type of student do you wish to enter?
       1. Undergrad
        2. Graduate
        3. Done
Choice: 1
Student name: Fred Duff
Student id: 12345
Number of credits: 14
Level: 2
What type of student do you wish to enter?

    Undergrad

       2. Graduate
       3. Done
Choice: 2
Student name: Sam Simmons
Student id: 45678
Number of credits: 14
Research Area: Life
What type of student do you wish to enter?

    Undergrad

        2. Graduate
       3. Done
Choice: 3
Undergrad Student:
       Name: Fred Duff
Id: 12345
       Credits: 14
Year Level: 2
Tuition: $7238.00
Grad Student:
       Name: Sam Simmons
       Id: 45678
       Credits: 14
       Research Area: Life
       Tuition: $9982.00
dkpvcs>
```

Submission

Zip your files and submit them to the HW 11 Assignment folder in MyCourses for this course.

Due Date

This assignment is due as listed in MyCourses.

Programming style - Header Comments – Fully follow coding standards - Method Comments – Describes what, not how it works - Variable names and comments – meaningful names and comments as needed - Block comments (loops and if logic) - White Space between code chunks (too much/too little) - Indentation SUBTOTAL StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - toString method written, uses motator - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - uses mutator to set attribute - calcTuition method - uses mutator to set attribute - calcTuition method	Points Deducted	Point Value	rogram 1
- Method Comments – Describes what, not how it works - Variable names and comments – meaningful names and comments as needed - Block comments (loops and if logic) - White Space between code chunks (too much/too little) - Indentation SUBTOTAL StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Tope use of inheritance - Attributes and constants properly declared - CalcTuition method - toString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - calls parameterized parent constructor - calls parameterized parent constructor - uses mutator to set attribute			rogramming style
- Variable names and comments – meaningful names and comments as needed - Block comments (loops and if logic) - White Space between code chunks (too much/too little) - Indentation SUBTOTAL StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method witten, uses mutator - CalcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Topper use of inheritance - Attributes and constants properly declared - Topper use of inheritance - Attributes and constants properly declared - Topper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - calls parameterized parent constructor - uses mutator to set attribute		5	
comments as needed Block comments (loops and if logic) White Space between code chunks (too much/too little) Indentation SUBTOTAL StudentTest Class Proper creation and use of Scanner Proper creation and use of ArrayList Proper loop for prompting user Proper loop for prompting user Proper casting/type testing Output formatted properly using printf() Student Class Attributes properly declared Default Constructor calls 3-arg constructor Parameterized constructor (3 args) uses mutators calcTuition method toString method uses accessors Froper use of inheritance Attributes and constants properly declared Default Constructor written, uses mutator Attributes and constants properly declared Default Constructor written, uses mutator Attributes and constants properly declared Default Constructor written, uses mutator Attributes and constants properly declared Default Constructor written, uses accessors UnderGrad Class Proper use of inheritance Attributes and constants properly declared ToString method written, uses accessors Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor calls parameterized parent constructor uses mutator to set attribute		5	Method Comments – Describes what, not how it works
- Block comments (loops and if logic) - White Space between code chunks (too much/too little) - Indentation SUBTOTAL StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		5	Variable names and comments – meaningful names and
- White Space between code chunks (too much/too little) - Indentation SUBTOTAL StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - toString method written, uses accessors - Proper use of inheritance - Caltributes and constants properly declared - Tull Parameterized Constructor - Calls parameterized Constructor - Calls parameterized parent constructor - Calls parameterized parent constructor - UnderGrad Class - Full Parameterized Constructor - Calls parameterized parent constructor - Calls parameterized parent constructor - UnderGrad Class - UnderGrad Class - Proper use of inheritance - Calls parameterized parent constructor		5	comments as needed
- Indentation 5 SUBTOTAL 30 StudentTest Class - Proper creation and use of Scanner 2 - Proper creation and use of ArrayList 3 - Proper loop for prompting user 3 - Proper use of for/each loop 4 - Proper casting/type testing 3 - Output formatted properly using printf() 3 Student Class - Attributes properly declared 2 - Default Constructor calls 3-arg constructor 3 - Parameterized constructor (3 args) uses mutators 4 - calcTuition method 3 - toString method uses accessors 4 Grad Class - Proper use of inheritance 2 - Attributes and constants properly declared 3 - Default Constructor written, uses mutator 4 - calcTuition method 3 - toString method written, uses mutator 4 - calcTuition method 3 - toString method written, uses mutator 4 - calcTuition method 3 - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance 2 - Attributes and constants properly declared 3 - Full Parameterized Constructor 3 - calls parameterized parent constructor 3 - calls parameterized parent constructor 3 - uses mutator to set attribute 2		5	Block comments (loops and if logic)
StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - calls parameterized parent constructor - uses mutator to set attribute		5	White Space between code chunks (too much/too little)
StudentTest Class - Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - calls parameterized parent constructor - uses mutator to set attribute		5	
- Proper creation and use of Scanner - Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Calls parameterized Constructor - Calls parameterized parent constructor - Calls parameterized parent constructor - Uses mutator to set attribute		30	UBTOTAL
- Proper creation and use of ArrayList - Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - CalcTuition method - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute			tudentTest Class
- Proper loop for prompting user - Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors 4 Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		2	- Proper creation and use of Scanner
- Proper use of for/each loop - Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - CalcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Calls parameterized Constructor - Calls parameterized parent constructor - Uses mutator to set attribute		3	- Proper creation and use of ArrayList
- Proper casting/type testing - Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses mutator - Attributes and constants properly declared - Attributes and constants properly declared - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Tull Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		3	 Proper loop for prompting user
- Output formatted properly using printf() Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		4	- Proper use of for/each loop
Student Class - Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		3	 Proper casting/type testing
- Attributes properly declared - Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors 4 UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - ToString method written, uses accessors - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		3	 Output formatted properly using printf()
- Default Constructor calls 3-arg constructor - Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute			tudent Class
- Parameterized constructor (3 args) uses mutators - calcTuition method - toString method uses accessors 4 Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		2	- Attributes properly declared
- calcTuition method - toString method uses accessors Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute		3	- Default Constructor calls 3-arg constructor
- toString method uses accessors Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute 4 4 4 5 6 7 8 9 9 10 10 10 10 10 10 10 10		4	- Parameterized constructor (3 args) uses mutators
Grad Class - Proper use of inheritance - Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute 2 - Attributes - Calls parameterized parent constructor - uses mutator to set attribute		3	- calcTuition method
 Proper use of inheritance Attributes and constants properly declared Default Constructor written, uses mutator calcTuition method toString method written, uses accessors UnderGrad Class Proper use of inheritance Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 		4	 toString method uses accessors
- Attributes and constants properly declared - Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute 3 - Attributes and constants properly declared - calls parameterized parent constructor - calls parameterized parent constructor			
- Default Constructor written, uses mutator - calcTuition method - toString method written, uses accessors UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute 4 2 4 4 Calls parameterized parent constructor 3 - uses mutator to set attribute		2	- Proper use of inheritance
 calcTuition method toString method written, uses accessors UnderGrad Class Proper use of inheritance Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 		3	 Attributes and constants properly declared
 toString method written, uses accessors UnderGrad Class Proper use of inheritance Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 		4	- Default Constructor written, uses mutator
UnderGrad Class - Proper use of inheritance - Attributes and constants properly declared - Full Parameterized Constructor - calls parameterized parent constructor - uses mutator to set attribute 2 2 3 • calls parameterized parent constructor 2		3	- calcTuition method
 Proper use of inheritance Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 		4	 toString method written, uses accessors
 Attributes and constants properly declared Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 			nderGrad Class
 Full Parameterized Constructor calls parameterized parent constructor uses mutator to set attribute 		2	- Proper use of inheritance
 calls parameterized parent constructor uses mutator to set attribute 		3	 Attributes and constants properly declared
• uses mutator to set attribute 2		3	- Full Parameterized Constructor
		3	 calls parameterized parent constructor
- calcTuition method 3		2	 uses mutator to set attribute
ı l		3	- calcTuition method
- toString method uses accessors 4		4	 toString method uses accessors
SUBTOTAL 70		70	
TOTAL: 100		100	OTAL:

Comments: