SOFTWARE QUALITY ASSURANCE LAB

ASSIGNMENT 3

Submitted by:
Shiv Kumar
2016UIT2563

Practical : Implement Lorenz and Kidd metric

Code :

```cpp
class Point{
        int x;
        int y;

        public:

        Point(int x, int y)
        {
                this->x = x;
                this->y = y;
        }

        void getPoint()
        {
                cout<<this->x<<" "<<this->y<<endl;
        }

        void setPoint(int x,int y)
        {
                this->x = x;
                this->y = y;
        }

        int getX(){
                return this->x;
        }

        int getY(){
                return this->y;
        }

};

class Shape{

        string color;

        public:

        string name;
        Shape(string name,string color){
```

```cpp
                setColor(color);
                setName(name);
        }

        string getColor(){
                return this->color;
        }

        string getName(){
                return this->name;
        }

        virtual void draw() = 0;

        virtual void setSize() = 0;

        virtual int getSize() = 0;

        private:
        string setColor(string color){
                this->color = color;
        }

        string setName(string name){
                this->name = name;
        }
};

class Circle : public Shape{

        Point center;
        int radius;
        int size;
        public:

        Circle(int x, int y, int radius){
                setCenter(x,y);
                setRadius(radius);
                setSize();
        }

        void draw()
        {
                cout<<center.getPoint()<<" "<<this->radius<<endl;
        }

        int getSize()
        {
                return this->size;
        }

        Point getCenter(){
```

```cpp
                return this->center;
        }

        int getRadius(){
                return this->radius;
        }

        private:
        void setSize()
        {
                this->size = this->radius * this->radius;
        }

        void setCenter(int x, int y){
                this->center = Point(x,y);
        }

        void setRadius(int radius)
        {
                this->radius = radius;
        }

};

class Square : public Shape{

        int width;
        int height;
        int size;

        public:

        Square(int width, int height){
                setWidth(width);
                setHeight(height);
                setSize();
        }

        void draw()
        {
                cout<<this->width<<" "<<this->height<<endl;
        }

        int getWidth()
        {
                return this->width;
        }

        int getHeight()
        {
                return this->height;
        }
```

```cpp
        int getSize()
        {
                return this->size;
        }

        private:
        void setSize()
        {
                this->size = this->height * this->width;
        }

        void setHeight(int height)
        {
                this->height = height;
        }

        void setWidth(int width)
        {
                this->width = width;
        }


};
```

Lorenz and Kidd metric for all classes :

1. **Point**
    1. **PM : 5**
    2. **NM : 5**
    3. **NPV : 0**
    4. **NV : 2**
    5. **NMI : 0**
    6. **NMO : 0**
    7. **NF : 0**
2. **Shape**
    1. **PM : 6**
    2. **NM : 8**
    3. **NPV : 1**
    4. **NV : 2**
    5. **NMI : 6**
    6. **NMO : 3**
    7. **NF : 0**
3. **Circle**
    1. **PM : 5**
    2. **NM : 8**
    3. **NPV : 0**
    4. **NV : 3**
    5. **NMI : 0**
    6. **NMO : 0**

7. **NF : 0**
4. **Square**
    1. **PM : 5**
    2. **NM : 8**
    3. **NPV : 0**
    4. **NV : 3**
    5. **NMI : 0**
    6. **NMO : 0**
    7. **NF : 0**