

Постановка задачи

Создание базового класса "Фигура" и его наследование для создания классов "Квадрат", "Прямоугольник" и "Круг". Класс "Фигура" будет иметь общие методы, такие как вычисление площади и периметра, а классы-наследники будут иметь специфичные методы и свойства.

Текст программы на Python:

```
# Создание базового класса "Фигура" и его наследование для создания классов
# "Квадрат", "Прямоугольник" и "Круг". Класс "Фигура" будет иметь общие методы,
# такие как вычисление площади и периметра, а классы-наследники будут иметь
# специфичные методы и свойства.

import math

# Родительский (супер) класс
class Figure:
    def __init__(self, side, width, height, radius): # Инициализация экземпляра класса
        self.side = side
        self.width = width
        self.height = height
        self.radius = radius

    def square_square(self):
        return self.side ** 2

    def perimetr_square(self):
        return self.side * 4

    def square_rectangle(self):
        return self.width * self.height

    def perimetr_rectangle(self):
        return (self.width + self.height) * 2

    def square_circle(self):
        return (self.radius ** 2) * math.pi

    def perimetr_circle(self):
        return 2 * math.pi * self.radius

# Классы Square, Rectangle и Circle - подклассы (дочерние классы)

# Определение площади и периметра Квадрата
class Square(Figure):
    def __init__(self, side):
        self.side = side

# Определение площади и периметра Прямоугольника
class Rectangle(Figure):

    def __init__(self, width, height):
        self.width = width
        self.height = height

# Определение площади и периметра Круга
class Circle(Figure):
```

```
def __init__(self, radius):  
    self.radius = radius  
  
a = Square(2)  
print(a.square_square())  
  
b = Rectangle(2, 2)  
print(b.square_rectangle())  
  
c = Circle(2)  
print(c.square_circle())
```

Протокол работы программы:

4

4

12.566370614359172