

Zadanie (Sahay)

Utwórz klasę `Shape`. Nie powinna ona posiadać żadnych pól danych. Powinna natomiast zawierać czysto wirtualną funkcję `get_area()`.

Utwórz klasę `Rectangle`, dziedziczącą po klasie `Shape`. Powinna ona posiadać dwa pola danych — jedno przechowujące szerokość prostokąta, a drugie jego wysokość. Oba pola powinny być typu `double`. Nadpisz funkcję `Shape::get_area()` w tej klasie. Ta nadpisana funkcja powinna zwracać *pole powierzchni* prostokąta. Dodaj również konstruktor do klasy.

Utwórz klasę `Ellipse`, również dziedziczącą po klasie `Shape`. Powinna ona mieć dwa pola danych — jedno przechowujące długość osi głównej (większej) elipsy, a drugie długość osi mniejszej. Oba pola powinny być typu `double`. Nadpisz funkcję `Shape::get_area()` w tej klasie. Ta funkcja powinna zwracać *pole powierzchni* elipsy. Dodaj także konstruktor do tej klasy.

Utwórz klasę `Canvas`. Nie powinna posiadać żadnych pól danych. Jej jedyna funkcja składowa, `display()`, powinna przyjmować *referencję* do obiektu typu `Shape` jako argument formalny. Wewnątrz funkcji `Canvas::display()` wywołaj metodę `Shape::get_area()` przy użyciu tej *referencji*.

Na koniec napisz funkcję `main()`, aby wykorzystać powyższe klasy. Zadeklaruj obiekty klas `Rectangle`, `Ellipse` oraz `Canvas`. Wywołaj funkcję `Canvas::display()` najpierw przekazując do niej obiekt klasy `Rectangle`, a następnie obiekt klasy `Ellipse`. Zaobserwuj wynik i ustal, czy została wywołana funkcja z klasy bazowej, czy z klasy pochodnej.

Następnie zdefiniuj ponownie funkcję `Shape::get_area()` jako funkcję *niewirtualną* i porównaj różnicę w wynikach działania programu.