

STT	Mã bài tập	Đề bài (Lớp và phương thức)	Test case	
			Input	Output
1	W8A1	<p>[Cylinder] Xây dựng lớp Cylinder (hình trụ) gồm: bán kính và chiều cao (kiểu số thực) với hai phương thức: tính diện tích bề mặt (diện tích toàn phần) và thể tích của hình trụ</p> <p>Đầu vào Đầu vào từ bàn phím gồm một dòng duy nhất chứa bán kính và chiều cao của hình trụ, cách nhau bởi một dấu cách.</p> <p>Đầu ra In ra màn hình diện tích bề mặt và thể tích của hình trụ (sử dụng lại những hàm vừa viết, $\pi=3.14$).</p> <p>(Kết quả làm tròn đến 2 chữ số sau dấu phẩy).</p>		
2	W8A2	<p>[Date] Xây dựng lớp Date biểu diễn ngày tháng năm với hai phương thức kiểm tra xem một ngày tháng nào đó có hợp lệ hay không và phương thức tính ngày kế tiếp của ngày tháng năm nào đó nếu nó là hợp lệ</p> <p>Nhập vào một ngày tháng năm, (ngày tháng năm biểu diễn dưới dạng dd/mm/yyyy). Nếu ngày tháng năm không tồn tại, in ra dòng chữ INVALID. Nếu là hợp lệ in ra ngày kế tiếp của nó.</p> <p>Đầu vào Đầu vào từ bàn phím chứa dữ liệu ngày, tháng, năm dưới dạng dd/mm/yyyy</p> <p>Đầu ra In ra màn hình ngày kế tiếp của ngày vừa nhập dưới dạng dd/mm/yyyy . Nếu ngày tháng không tồn tại, in ra INVALID .</p>		

3	W8A3	<p>[Point] Xây dựng lớp điểm với hai thuộc tính x, y cùng các phương thức kiểm tra xem có phải gốc tọa độ không, có nằm trên trục tung không, có nằm trên trục hoành không, tính khoảng cách từ điểm đó tới một điểm khác</p> <p>Đầu vào: Nhập vào tọa độ một điểm M bất kỳ</p> <p>Đầu ra: Thông báo M có phải gốc tọa độ không hoặc có nằm trên trục tung hoặc trục hoành không, khoảng cách từ M tới gốc tọa độ là bao nhiêu (lấy 2 chữ số phần thập phân)</p>		
4	W8A4	<p>[Calculator] Xây dựng lớp Calculator (máy tính bỏ túi) với các phương thức để thực hiện các phép toán 1. add (cộng), 2. subtract(trừ), 3. product (nhân) và 4. divide(chia), 5. power(hàm mũ), 6. mod (lấy phần dư) và phương thức 7. set_numbers(cập nhật 2 số mới) nếu người dùng muốn tính toán trên 2 số khác.</p> <p>Khởi tạo lớp là hai thuộc tính gồm 2 số cần tính, Mỗi phương thức thực hiện một phép tính và trả về kết quả tương ứng</p> <p>Tạo menu và cho người dùng tính toán tới khi gõ "yes", "YES" hoặc "Yes" cho câu hỏi "Bạn có muốn thoát?" mới dừng.</p>		

5	W8A5	<p>[ShoppingCart] Xây dựng lớp ShoppingCart (giỏ hàng) để quản lý các sản phẩm của khách. Dùng danh sách để lưu các sản phẩm, mỗi sản phẩm là một từ điển có name (tên) và price(giá). Với các phương thức sau:</p> <ol style="list-style-type: none"> 1. Khởi tạo với danh sách rỗng 2. Thêm vào giỏ hàng một sản phẩm mới 3. Xóa khỏi giỏ một sản phẩm nào đó (tìm theo tên để xóa) 4. Kiểm tra xem giỏ hàng có rỗng không 5. Tính tổng tiền giỏ hàng 6. Hiển thị toàn bộ sản phẩm trong giỏ 7. Xóa toàn bộ giỏ hàng <p>Tạo menu cho người dùng nhập lựa chọn truy vấn từ 2 tới 7. Chương trình dừng khi người dùng nhập "yes" cho câu hỏi "Có muốn thoát khỏi chương trình không?"</p>		
6	W8A6	<p>Yêu cầu Xây dựng lớp Rectangle(w, h) với các phương thức:</p> <p>area() trả về diện tích.</p> <p>perimeter() trả về chu vi.</p> <p>scale(k) phóng to hình chữ nhật theo hệ số k (nhân cả w và h), không trả về gì.</p> <p>Input w h k (số thực dương)</p> <p>Output area perimeter sau khi phóng to (làm tròn 2 chữ số thập phân).</p>	<p>2 5 2</p> <p>10 0.5 3</p> <p>1.2 1.8 0.5</p>	<p>40.00 28.00</p> <p>45.00 63.00</p> <p>0.54 3.00</p>

7	W8A7	<p>Yêu cầu Lớp Fraction(num, den) (tử, mẫu nguyên, mẫu ≠ 0) với:</p> <p>Rút gọn phân số trong __init__ (quy ước mẫu > 0).</p> <p>Các phương thức add/sub/mul/div trả về phân số đã rút gọn.</p> <p>__str__() in dạng a/b.</p> <p>Input a b op c d (tương ứng a/b (op) c/d, op thuộc {+, -, *, /})</p> <p>Output Phân số kết quả đã rút gọn</p>	<p>2 4 - 1 4</p> <p>-2 3 * 3 5</p> <p>7 8 / 14 3</p> <p>1 2 + 1 3</p>	<p>1/4</p> <p>-2/5</p> <p>3/16</p> <p>5/6</p>
8	W8A8	<p>Yêu cầu Lớp BankAccount(owner, balance) với:</p> <p>deposit(x), withdraw(x); nếu rút quá số dư thì bỏ qua giao dịch.</p> <p>__str__() in owner: balance.</p> <p>Input Dòng 1: owner initial_balance Dòng 2: n (số lệnh) Tiếp theo n dòng, mỗi dòng: DEPOSIT x hoặc WITHDRAW x.</p> <p>Output Số dư cuối cùng (in số thực, 2 chữ số).</p>	<p>Alice 100 3 DEPOSIT 50 WITHDRAW 120 WITHDRAW 20</p> <p>Bob 0 4 WITHDRAW 10 DEPOSIT 10 WITHDRAW 5 WITHDRAW 10</p> <p>Kim 200 2 DEPOSIT 0.5 WITHDRAW 50.25</p> <p>Lan 10 1 WITHDRAW 10</p>	<p>10.00</p> <p>5.00</p> <p>150.25</p> <p>0.00</p>
		<p>Yêu cầu Lớp Student(name) với:</p> <p>add_score(subject, score)</p>	<p>Minh 3 Math 9 Phys 8 Eng 7.5</p>	<p>Minh 8.17 Excellent</p>

9	W8A9	add_score(subject, score) avg() trả về điểm trung bình. rank() trả về "Excellent" nếu avg ≥ 8, "Good" nếu 6.5–<8, "Average" nếu 5–<6.5, ngược lại "Poor".	Ha 2 A 6 B 7	Ha 6.50 Good
		Input Dòng 1: name Dòng 2: n n dòng tiếp: subject score (score là số thực 0–10)	Nam 3 X 4 Y 5 Z 6	Nam 5.00 Average
		Output name avg rank (avg 2 chữ số).	KieuAnh 1 Only 3.4	KieuAnh 3.40 Poor
10	W8A10	Yêu cầu Lớp Book(title, author, year). Lớp Library() với: add(book) count_by_author(name) find_by_year(y) trả về số sách xuất bản năm y.	3 ADD A;X;2000 ADD B;X;2001 COUNT X	2
		Input Dòng 1: n (số thao tác) n dòng: ADD title;author;year COUNT author (đếm theo tác giả) COUNTYEAR year (đếm theo năm)	4 ADD T;A;1999 ADD U;B;1999 ADD V;A;2000 COUNTYEAR 1999	2
		Output Mỗi lệnh COUNT/COUNTYEAR in một dòng kết quả.	2 ADD Only;Me;2020 COUNT You	0

11	W8A11	<p>Bạn được cho bốn điểm A, B, C và D trong hệ tọa độ Đề-các 3 chiều.</p> <p>Nhiệm vụ của bạn là tính và in ra góc giữa hai mặt phẳng tạo bởi các điểm:</p> <p>mặt phẳng (A, B, C)</p> <p>mặt phẳng (B, C, D)</p> <p>Góc cần in ra được ký hiệu là PHI, và phải dùng độ (degrees), không dùng radian.</p> <p>Dạng Input:</p> <p>Một dòng input chứa 3 số thực, ngăn cách bằng dấu cách, biểu diễn:</p> <p>tọa độ X tọa độ Y tọa độ Z của một điểm.</p> <p>Dạng Output:</p> <p>In ra góc giữa hai mặt phẳng, chính xác tới hai chữ số thập phân.</p>	<p>0 4 5</p> <p>1 7 6</p> <p>0 5 9</p> <p>1 7 2</p> <p>5 8.8 9</p> <p>4 -1 3</p> <p>7 8.7 3.3</p> <p>4.4 5.1 6.3</p>	<p>8.19</p> <p>5.69</p>
12	W8A12	<p>[Complex]</p> <p>Khai báo kiểu dữ liệu biểu diễn số phức \ (ComplexNumber\) bao gồm \ (2\) biến \ (realPart\) và \ (imaginaryPart\) kiểu \ (double\) biểu diễn phần thực và phần ảo của một số phức.</p> <p>Viết hàm khởi tạo ComplexNumber(double initRealPart, double initImaginaryPart) để khởi tạo các giá trị phần thực và phần ảo của một số phức và hàm ComplexNumber multiply(ComplexNumber other) trả về tích của số phức hiện tại và một số phức khác.</p> <p>Đầu vào</p> <p>Đầu vào từ bàn phím gồm hai dòng, mỗi dòng gồm hai số thực cách nhau bởi một dấu cách biểu diễn phần thực và phần ảo của một số phức. Các số trong đầu vào có giá trị tuyệt đối không vượt quá \ (100\).</p> <p>Đầu ra</p> <p>In ra màn hình hai số biểu diễn phần thực và phần ảo của tích hai số phức.</p>	<p>1 2</p> <p>3 4</p> <p>5 6</p> <p>7 8</p> <p>10 1</p> <p>0 2</p>	<p>-5 10</p> <p>-13 82</p> <p>-2 20</p>

13	W8A13	<p>[Triangle]</p> <p>Khai báo cấu trúc dữ liệu tam giác <code>\(Triangle\)</code> chứa <code>\(3\)</code> biến <code>\(egde1,\,edge2,\,edge3\)</code> kiểu <code>\(int\)</code> là độ dài <code>\(3\)</code> cạnh của tam giác đó.</p> <p>Viết hàm khởi tạo <code>Triangle(int initEdge1,int initEdge2, int initEdge3)</code> để khởi tạo độ dài các cạnh của tam giác và hàm <code>double getArea()</code> trả về diện tích của tam giác đó (nếu tam giác không hợp lệ, trả về <code>\(-1\)</code>).</p> <p>Hoàn thiện chương trình dưới đây nhập vào <code>\(3\)</code> cạnh của một tam giác và in ra diện tích của tam giác đó.</p> <p>Đầu vào</p> <p>Một dòng duy nhất từ bàn phím chứa <code>\(3\)</code> số nguyên là độ dài <code>\(3\)</code> cạnh của tam giác</p> <p>Đầu ra</p> <p>In ra màn hình diện tích tam giác (làm tròn đến <code>\(2\)</code> chữ số thập phân sau dấu phẩy). In ra <code>\(invalid\)</code> nếu <code>\(3\)</code> số nhập vào không phải là <code>\(3\)</code> cạnh của một tam giác</p>	<p>3 4 5</p> <p>1 2 3</p> <p>5 6 7</p>	<p>6</p> <p>invalid</p> <p>14.70</p>
----	-------	---	--	--------------------------------------