

STT	Mã bài tập	Đề bài (Cấu trúc điều khiển vòng lặp)	Test case	
			Input	Output
1	W4A1	[SumOfNumBers] Viết chương trình Python sử dụng một trong 2 loại vòng lặp for hoặc while để tính tổng các số nguyên từ 1 đến n. (n <= 1000)	3 5 100 1000 50	6 15 5050 500500 1275
2	W4A2	[PrimeNumber] Viết chương trình nhập vào một số tới khi được một số nguyên dương thì thôi. Kiểm tra số này có phải là số nguyên tố hay không?	4 7 13 9 100	False True True False False
3	W4A3	[Factorial] Viết chương trình tính giai thừa của một số nguyên n cho trước ($n! = 1 \times 2 \times \dots \times n$) với $0 < n < 100$.	\	6 24 1 120 40320
4	W4A4	[CountingDigits] Viết chương trình nhập vào một số nguyên n và in ra màn hình số chữ số (trừ dấu) của số đó. Không dùng kiểu dữ liệu string.	234 -1593 1234567 -1223240 243430	3 4 7 7 6

5	W4A5	<p>[MeaningOfLife]</p> <p>Khi Douglas Adam viết cuốn sách, The Hitchhiker's Guide to the Galaxy, ông ta đã thêm vào đó một vài lời nói đùa, và theo năm tháng nó đã trở nên nổi tiếng hơn cả bản thân cuốn tiểu thuyết: "Câu trả lời cho câu hỏi lớn nhất về sự sống, về vũ trụ và tất cả mọi thứ là 42." Một vài người cuồng tín đã tốn hàng năm trời, với mọi cố gắng nỗ lực để gán ghép ý nghĩa, biểu trưng cho con số 42 và các "sự cố" liên quan đến nó.</p> <p>Với mong muốn hái ra tiền từ sự "ám ảnh" điên rồ này, một cuốn sách mới vừa được xuất bản, 42: Douglas Adams' Amazingly Accurate Answer to Life, the Universe and Everything. Cuốn sách được phát hành trùng với kỉ niệm 10 năm ngày mất của Adams.</p> <p>"Câu trả lời rất đơn giản", như Adams đã nói, "Chỉ là một trò đùa. Tôi cần một con số, một con số bé bé, một con số tầm thường, và tôi đã chọn nó. Hai chữ số, bội của 13, một con số hoàn toàn vô nghĩa đối với mấy tay thầy chùa. Tôi ngồi trên bàn làm việc, ngắm khu vườn và nghĩ rằng số 42 sẽ được việc lắm đây. Tôi gõ nó ra. Và hết."</p> <p>Xuyên suốt lịch sử, các con số được gán cho những ý nghĩa đặc biệt nào đó. Plato gọi việc nghiên cứu các biểu tượng số học" là level cao nhất của tri thức. Trong khi Pythagoras tin rằng các con số đều có linh hồn cũng như sức mạnh kì diệu. Trong khi đó, hàng triệu người hâm mộ Hitchhiker cho đến ngày nay vẫn kiên trì giải mã những thứ họ cho rằng đó là những động cơ bí mật của Adams.</p> <p>Hãy viết một chương trình nhận vào một số nguyên dương n và một dãy gồm n số nguyên. Nếu n số nguyên đó có số 42 thì in ra màn hình dòng chữ "I've found the meaning of life!", ngược lại in ra dòng chữ "It's a joke!".</p>	5 1 192 32 42 39 2 3 4 5 54 23 12 34 42 10 1 2 3 4 4 5 6 7 8 9 7 42 54 6 7 8 12 89	I've found the meaning of life! It's a joke! I've found the meaning of life! It's a joke! I've found the meaning of life!
6	W4A6	<p>"[SumPrimesInRange]</p> <p>Nhập hai số nguyên dương a, b ($a \leq b$). Hãy tính tổng các số nguyên tố trong đoạn [a, b]. Mỗi dòng là một bộ test (a b)."</p>	2 10 1 1 14 16 10 20 100 120	17 0 0 60 533
7	W4A7	<p>"[LargestPrimeFactor]</p> <p>Nhập vào số nguyên dương n ($n \geq 2$). Hãy in ra ước số nguyên tố lớn nhất của n."</p>	10 27 97 64 99991	5 3 97 2 99991

8	W4A8	ReverseAddPalindrome] Nhập số nguyên dương n. Lập lại phép toán $n = n + \text{reverse}(n)$ cho đến khi n là số đối xứng (palindrome). In ra số bước và giá trị palindrome thu được.	56 57 87 89 195	1 121 2 363 4 4884 24 8813200023188 4 9339
9	W4A9	[PerfectSquaresDistinctDigits] Nhập vào số nguyên dương n. In ra tất cả các số chính phương $\leq n$ có các chữ số đôi một khác nhau (in theo thứ tự tăng dần, cùng hàng, cách nhau bởi một dấu cách). Nếu không có, in ra (no number).	10 50 100 1000 5000	1 4 9 1 4 9 16 25 36 49 1 4 9 16 25 36 49 64 81 1 4 9 16 25 36 49 64 81 169 196 256 1 4 9 16 25 36 49 64 81 169 196 256
10	W4A10	"[LongestCollatzUpToN] Nhập vào số nguyên dương n. Tìm số x trong $[1, n]$ có độ dài dãy Collatz lớn nhất và in ra x cùng độ dài dãy tương ứng (x L). Nếu có nhiều số cùng độ dài, chọn số nhỏ nhất."	10 20 50 100 200	9 20 18 21 27 112 97 119 171 125
11	W4A11	[EvenDivisors] Viết chương trình đếm số lượng ước số chẵn của số nguyên dương n với $n < 10^6$.	8 17 1994 2048 10024	3 0 2 11 12
12	W4A12	[Interest] Một người có tài khoản tiết kiệm ở ngân hàng và gửi vào X đồng với lãi suất là 0.7% mỗi tháng. Viết chương trình tính số tiền sau N tháng người ấy rút được (cả gốc và lãi, bỏ qua phần lẻ thập phân).	1000 12 1999 4 2020 0 200000 12 7800 48	1087 2055 2020 217462 10902
13	W4A13	[ClosePairNumber] Hai số nguyên dương tạo thành một cặp số thân thiết khi chúng tuân theo quy luật: Số này bằng tổng tất cả các ước của số kia (trừ chính nó) và ngược lại. Ví dụ 220 và 284 là một cặp số thân thiết 220 ngoài chính nó ra, nó còn có 11 ước số khác là: 1, 2, 4, 5, 10, 11, 20, 44, 55, 110. Tổng của 11 ước này cho chúng ta 284 284 ngoài bản thân nó, còn có 5 ước số khác là: 1, 2, 4, 71, 142. Tổng của chúng vừa đúng bằng 220 Nhập hai số từ bàn phím và kiểm tra xem chúng có phải là cặp số thân thiết không. Nếu có, in ra màn hình "true", ngược lại, in ra "false"	220 284 10 7 1992 2012 1184 1210 2620 2924	true false false true true

14	W4A14	[GCD] Cho hai số m và n là hai số nguyên dương nhập từ bàn phím. Viết chương trình tìm ước chung lớn nhất của hai số m và n.	3 1 15 42 2048 197 19876 248 100024 1024	1 3 1 4 8
15	W4A15	[ChickenAndDog] "Vừa gà vừa chó, Bó lại cho tròn, Ba mươi sáu con, Một trăm chân chẵn" Từ bài toán dân gian trên, mở rộng thành chương trình nhận đầu vào là tổng số con và tổng số chân của gà và chó. Nếu tìm được số phù hợp, in ra số lượng gà và số lượng chó. Ngược lại, in ra "invalid".	36 100 52 117 26 67 78 194 120 242	22 14 invalid invalid 59 19 119 1
	W4A16	In các số chẵn chia hết cho 3 bé hơn 100		
	W4A17	Nhập vào số nguyên dương a, in ra bảng cửu chương của a		
	W4A18	Nhập vào số nguyên dương a và b, in toàn bộ ước chung của a và b		
	W4A19	[EvenNumBers] Viết chương trình Python sử dụng một trong 2 loại vòng lặp for hoặc while để in ra các số dương chẵn nhỏ hơn hoặc bằng n (n <= 1000)	8 20 3 9 10	2 4 6 8 2 4 6 8 10 12 14 16 18 20 2 2 4 6 8 2 4 6 8
	W4A20	Nhập vào một số nguyên dương n, kiểm tra xem n có phải là số dạng 2^k hay không		
	W4A21	Nhập vào số nguyên dương n, tính tổng các chữ số của n		
	W4A22	Nhập vào số nguyên dương n, đếm xem n có bao nhiêu chữ số chẵn, bao nhiêu chữ số lẻ		
	W4A23	Nhập n Cho $S(k) = 1 + 2 + 3 + \dots + k$ Tìm k sao cho S(k) lớn nhất nhưng nhỏ hơn n		
	W4A24	Nhập vào A, tìm n nhỏ nhất sao cho $1 + 1/2 + 1/3 + 1/4 + \dots + 1/n > A$		
	W4A25	Nhập vào một dãy số nguyên, ngưng nhập khi người dùng nhập -1. Sau khi nhập xong, in số lớn nhất, số nhỏ nhất trong những số vừa nhập		

W4A26	<p>[Fibonacci]</p> <p>Dãy số fibonacci là dãy số được định nghĩa như sau: 1, 1, 2, 3, 5, 8, 13,... với số kế tiếp sẽ bằng tổng hai số trước đó</p> <p>Nhập vào số nguyên dương A, hãy tìm số trong dãy số fibonacci lớn nhất nhưng không vượt quá A</p>		
W4A27	<p>Nhập vào một chuỗi, hãy đếm xem trong chuỗi có bao nhiêu từ (quy định là chuỗi không có ký tự đặc biệt, không số, không có dấu câu, chỉ có ký tự chữ và khoảng trắng)</p> <p>Dùng cách: for trên index; KHÔNG dùng hàm split(), len()</p>		
W4A28	<p>Nhập vào một chuỗi, hãy in từ đầu tiên trong chuỗi</p> <p>Dùng cách: for trên index; KHÔNG dùng hàm split()</p>		
W4A29	<p>Nhập vào một chuỗi có dạng 3 số nguyên, mỗi số nguyên cách nhau một dấu phẩy, hãy tính tổng 3 số nguyên đó</p> <p>VD:</p> <p>Nhập: 3, 12, 15</p> <p>Tổng: 30</p>		
W4A30	<p>Nhập vào một chuỗi, hãy đếm xem trong chuỗi có bao nhiêu ký tự in hoa, bao nhiêu ký tự in thường, bao nhiêu ký tự số</p>		
W4A31	<p>Nhập vào một chuỗi, hãy tách toàn bộ ký tự số trong chuỗi ra rồi tính tổng của chúng</p> <p>VD:</p> <p>Nhập chuỗi: abd45ecf47wde3s1</p> <p>Tổng: $4 + 5 + 4 + 7 + 3 + 1 = 24$</p>		
W4A32	<p>Nhập vào một chuỗi, kiểm tra chuỗi đó có phải là một chuỗi mật khẩu mạnh hay không (một chuỗi mật khẩu mạnh cần có ít nhất 1 ký tự đặc biệt, 1 ký tự in hoa, 1 con số, 1 chữ thường và độ dài phải lớn hơn 6)</p>		
W4A33	<p>Nhập vào một số nguyên, hãy chuyển số sang chuỗi, rồi đặt dấu chấm phân tách mỗi 3 chữ số (phân cách phần ngàn) rồi in ra màn hình</p> <p>VD:</p> <p>Nhập số: 375469485</p> <p>Đổi sang chuỗi rồi in ra: 375.469.485</p>		
W4A34	<p>Nhập vào chuỗi a và chuỗi b</p> <p>Hãy xóa chuỗi b trong chuỗi a rồi in lại chuỗi a ra màn hình (không dùng hàm replace)</p> <p>Ví dụ:</p> <p>Chuỗi a: "Xin chào mọi người!"</p> <p>Chuỗi b: "Xin chào"</p> <p>Sau khi xóa, chuỗi a: " mọi người!"</p>		