

Status	Finished
Started	Sunday, 12 January 2025, 2:50 AM
Completed	Sunday, 12 January 2025, 2:50 AM
Duration	6 secs
Marks	0.00/110.00
Grade	0.00 out of 10.00 (0%)

Question 1

Not answered

Mark 0.00 out of 10.00

[BasicLoop-SmallerThanN]

Nhập vào số nguyên dương  $n$  và in ra dãy các số dương nhỏ hơn hoặc bằng  $n$ , cách nhau bởi dấu cách.

For example:

Input	Result
3	1 2 3

Answer: (penalty regime: 0 %)

1	
---	--

## Question 2

Not answered

Mark 0.00 out of 10.00

### [NthFibonacciNumber]

[Dãy số Fibonacci](#) được Fibonacci, một nhà toán học người Ý, công bố vào năm 1202 trong cuốn sách Liber Abacci. [Dãy số Fibonacci](#) được phát biểu dưới dạng bài toán con thỏ như sau:

Một đôi thỏ (gồm một thỏ đực và một thỏ cái) cứ mỗi tháng đẻ được một đôi thỏ con (cũng gồm một thỏ đực và thỏ cái); một đôi thỏ con, khi tròn 2 tháng tuổi, sau mỗi tháng đẻ ra một đôi thỏ con, và quá trình sinh nở cứ thế tiếp diễn. Hỏi  $n$  tháng bao nhiêu đôi thỏ, nếu tháng đầu tiên có một đôi thỏ sơ sinh?

Ta nhận thấy,

- Tháng đầu tiên và tháng thứ hai chỉ có 1 đôi thỏ.
- Tháng thứ ba: đôi thỏ này sẽ sinh ra một đôi thỏ con, do đó trong tháng này có 2 đôi thỏ.
- Tháng thứ tư: chỉ có đôi thỏ ban đầu sinh con nên đến thời điểm này có 3 đôi thỏ.
- Tháng thứ năm: có hai đôi thỏ (đôi thỏ đầu và đôi thỏ được sinh ra ở tháng Ba) cùng sinh con nên ở tháng này có  $2 + 3 = 5$  đôi thỏ.
- ...

Khái quát, nếu  $n$  là số nguyên dương, gọi  $f(n)$  là số đôi thỏ có ở tháng thứ  $n$ , ta có:

- Với  $n \leq 2$ ,  $f[n] = 1$ .
- Với  $n > 2$ ,  $f[n] = f[n - 1] + f[n - 2]$ .

Viết chương trình tính số Fibonacci thứ  $n$ .

#### Đầu vào

Đầu vào từ bàn phím, chứa duy nhất số nguyên dương  $n$  ( $n \leq 30$ ).

#### Đầu ra

In ra màn hình số Fibonacci thứ  $n$ .

**For example:**

Input	Result
5	5

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 3**

Not answered

Mark 0.00 out of 10.00

**[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"

Đầu vào: Hai số nguyên lần lượt là tổng số con và tổng số chân của [gà và chó](#).

Đầu ra: Số lượng gà và số lượng chó phù hợp. Nếu không có, in ra "invalid".

**Dữ liệu vào nhập từ bàn phím và kết quả được in ra màn hình.**

**For example:**

Input	Result
36 100	chicken = 22, dog = 14

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 4**

Not answered

Mark 0.00 out of 10.00

**[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.

**For example:**

Input	Result
12	2
-1593	4

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 5**

Not answered

Mark 0.00 out of 10.00

**[DivideBySeven]**

Nhập một số nguyên dương  $n$  từ bàn phím (  $n < 10000$ ). In ra màn hình tất cả các số nguyên nhỏ hơn hoặc bằng  $n$  và chia hết cho 7.

**For example:**

Input	Result
15	7 14

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 6**

Not answered

Mark 0.00 out of 10.00

**[FivePerLine]**

Viết chương trình chỉ sử dụng một vòng lặp "for" và một câu lệnh điều kiện "if" để in ra màn hình các số nguyên từ 1000 đến 2000. Biết rằng mỗi dòng chứa đúng 5 số nguyên.

**Gợi ý:** sử dụng toán tử %

**For example:**

Result					
1000	1001	1002	1003	1004	
1005	1006	1007	1008	1009	
1010	1011	1012	1013	1014	
1015	1016	1017	1018	1019	
1020	1021	1022	1023	1024	
1025	1026	1027	1028	1029	
1030	1031	1032	1033	1034	
1035	1036	1037	1038	1039	
1040	1041	1042	1043	1044	
1045	1046	1047	1048	1049	
1050	1051	1052	1053	1054	
1055	1056	1057	1058	1059	
1060	1061	1062	1063	1064	
1065	1066	1067	1068	1069	
1070	1071	1072	1073	1074	
1075	1076	1077	1078	1079	
1080	1081	1082	1083	1084	
1085	1086	1087	1088	1089	
1090	1091	1092	1093	1094	
1095	1096	1097	1098	1099	
1100	1101	1102	1103	1104	
1105	1106	1107	1108	1109	
1110	1111	1112	1113	1114	
1115	1116	1117	1118	1119	
1120	1121	1122	1123	1124	
1125	1126	1127	1128	1129	
1130	1131	1132	1133	1134	
1135	1136	1137	1138	1139	
1140	1141	1142	1143	1144	
1145	1146	1147	1148	1149	
1150	1151	1152	1153	1154	
1155	1156	1157	1158	1159	
1160	1161	1162	1163	1164	
1165	1166	1167	1168	1169	
1170	1171	1172	1173	1174	
1175	1176	1177	1178	1179	
1180	1181	1182	1183	1184	
1185	1186	1187	1188	1189	
1190	1191	1192	1193	1194	
1195	1196	1197	1198	1199	
1200	1201	1202	1203	1204	
1205	1206	1207	1208	1209	
1210	1211	1212	1213	1214	
1215	1216	1217	1218	1219	
1220	1221	1222	1223	1224	
1225	1226	1227	1228	1229	
1230	1231	1232	1233	1234	
1235	1236	1237	1238	1239	
1240	1241	1242	1243	1244	
1245	1246	1247	1248	1249	
1250	1251	1252	1253	1254	
1255	1256	1257	1258	1259	
1260	1261	1262	1263	1264	
1265	1266	1267	1268	1269	
1270	1271	1272	1273	1274	

Result
1275 1276 1277 1278 1279
1280 1281 1282 1283 1284
1285 1286 1287 1288 1289
1290 1291 1292 1293 1294
1295 1296 1297 1298 1299
1300 1301 1302 1303 1304
1305 1306 1307 1308 1309
1310 1311 1312 1313 1314
1315 1316 1317 1318 1319
1320 1321 1322 1323 1324
1325 1326 1327 1328 1329
1330 1331 1332 1333 1334
1335 1336 1337 1338 1339
1340 1341 1342 1343 1344
1345 1346 1347 1348 1349
1350 1351 1352 1353 1354
1355 1356 1357 1358 1359
1360 1361 1362 1363 1364
1365 1366 1367 1368 1369
1370 1371 1372 1373 1374
1375 1376 1377 1378 1379
1380 1381 1382 1383 1384
1385 1386 1387 1388 1389
1390 1391 1392 1393 1394
1395 1396 1397 1398 1399
1400 1401 1402 1403 1404
1405 1406 1407 1408 1409
1410 1411 1412 1413 1414
1415 1416 1417 1418 1419
1420 1421 1422 1423 1424
1425 1426 1427 1428 1429
1430 1431 1432 1433 1434
1435 1436 1437 1438 1439
1440 1441 1442 1443 1444
1445 1446 1447 1448 1449
1450 1451 1452 1453 1454
1455 1456 1457 1458 1459
1460 1461 1462 1463 1464
1465 1466 1467 1468 1469
1470 1471 1472 1473 1474
1475 1476 1477 1478 1479
1480 1481 1482 1483 1484
1485 1486 1487 1488 1489
1490 1491 1492 1493 1494
1495 1496 1497 1498 1499
1500 1501 1502 1503 1504
1505 1506 1507 1508 1509
1510 1511 1512 1513 1514
1515 1516 1517 1518 1519
1520 1521 1522 1523 1524
1525 1526 1527 1528 1529
1530 1531 1532 1533 1534
1535 1536 1537 1538 1539
1540 1541 1542 1543 1544
1545 1546 1547 1548 1549



Result
1550 1551 1552 1553 1554
1555 1556 1557 1558 1559
1560 1561 1562 1563 1564
1565 1566 1567 1568 1569
1570 1571 1572 1573 1574
1575 1576 1577 1578 1579
1580 1581 1582 1583 1584
1585 1586 1587 1588 1589
1590 1591 1592 1593 1594
1595 1596 1597 1598 1599
1600 1601 1602 1603 1604
1605 1606 1607 1608 1609
1610 1611 1612 1613 1614
1615 1616 1617 1618 1619
1620 1621 1622 1623 1624
1625 1626 1627 1628 1629
1630 1631 1632 1633 1634
1635 1636 1637 1638 1639
1640 1641 1642 1643 1644
1645 1646 1647 1648 1649
1650 1651 1652 1653 1654
1655 1656 1657 1658 1659
1660 1661 1662 1663 1664
1665 1666 1667 1668 1669
1670 1671 1672 1673 1674
1675 1676 1677 1678 1679
1680 1681 1682 1683 1684
1685 1686 1687 1688 1689
1690 1691 1692 1693 1694
1695 1696 1697 1698 1699
1700 1701 1702 1703 1704
1705 1706 1707 1708 1709
1710 1711 1712 1713 1714
1715 1716 1717 1718 1719
1720 1721 1722 1723 1724
1725 1726 1727 1728 1729
1730 1731 1732 1733 1734
1735 1736 1737 1738 1739
1740 1741 1742 1743 1744
1745 1746 1747 1748 1749
1750 1751 1752 1753 1754
1755 1756 1757 1758 1759
1760 1761 1762 1763 1764
1765 1766 1767 1768 1769
1770 1771 1772 1773 1774
1775 1776 1777 1778 1779
1780 1781 1782 1783 1784
1785 1786 1787 1788 1789
1790 1791 1792 1793 1794
1795 1796 1797 1798 1799
1800 1801 1802 1803 1804
1805 1806 1807 1808 1809
1810 1811 1812 1813 1814
1815 1816 1817 1818 1819
1820 1821 1822 1823 1824

Result
1825 1826 1827 1828 1829
1830 1831 1832 1833 1834
1835 1836 1837 1838 1839
1840 1841 1842 1843 1844
1845 1846 1847 1848 1849
1850 1851 1852 1853 1854
1855 1856 1857 1858 1859
1860 1861 1862 1863 1864
1865 1866 1867 1868 1869
1870 1871 1872 1873 1874
1875 1876 1877 1878 1879
1880 1881 1882 1883 1884
1885 1886 1887 1888 1889
1890 1891 1892 1893 1894
1895 1896 1897 1898 1899
1900 1901 1902 1903 1904
1905 1906 1907 1908 1909
1910 1911 1912 1913 1914
1915 1916 1917 1918 1919
1920 1921 1922 1923 1924
1925 1926 1927 1928 1929
1930 1931 1932 1933 1934
1935 1936 1937 1938 1939
1940 1941 1942 1943 1944
1945 1946 1947 1948 1949
1950 1951 1952 1953 1954
1955 1956 1957 1958 1959
1960 1961 1962 1963 1964
1965 1966 1967 1968 1969
1970 1971 1972 1973 1974
1975 1976 1977 1978 1979
1980 1981 1982 1983 1984
1985 1986 1987 1988 1989
1990 1991 1992 1993 1994
1995 1996 1997 1998 1999
2000

Answer: (penalty regime: 0 %)

1	
---	--

### Question 7

Not answered

Mark 0.00 out of 10.00

#### [GCD]

Cho hai số  $m$  và  $n$  là hai số nguyên nhập từ bàn phím. Viết chương trình tìm ước chung lớn nhất của hai số  $m$  và  $n$  ( $UCLN > 0$ ).

Lưu ý: Xử lý cả trường hợp  $m, n < 0$ .

Đầu vào: Hai số nguyên  $m$  và  $n$ .

Đầu ra: Ước chung lớn nhất (ước dương) của hai số.

**Dữ liệu vào nhập từ bàn phím và kết quả được in ra màn hình.**

**For example:**

Input	Result
3 1	1

**Answer:** (penalty regime: 0 %)

1

**Question 8**

Not answered

Mark 0.00 out of 10.00

**[OddNumbers]**

**Tính chẵn lẻ** là một thuật ngữ toán học mô tả đặc tính của một số nguyên có thể thuộc về một trong hai nhóm: chẵn hoặc lẻ. [Số chẵn](#) là một số nguyên chia hết cho 2 và **số lẻ** là một số nguyên không phải là [số chẵn](#). Chẳng hạn số 0 là một [số chẵn](#). Tính chẵn lẻ không áp dụng cho các số không phải là số nguyên.

Viết chương trình nhận vào số nguyên  $n$  và in dãy gồm  $n$  số lẻ cách nhau bởi dấu cách (dãy bắt đầu từ 1).

**For example:**

Input	Result
2	1 3

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 9**

Not answered

Mark 0.00 out of 10.00

**[FirstNFiboNumbers]**

Dãy Fibonacci mang tên của Fibonacci, còn gọi là Leonardo xứ Pisa hay Leonardo Pisano, các số Fibonacci lần đầu tiên được nêu ra trong tác phẩm *Liber abaci* của ông vào năm 1202.

Dãy Fibonacci là một dãy số trong đó một con số được xác định bằng cách cộng hai con số đứng trước nó. Bắt đầu với 0 và 1, dãy số tiếp tục 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, và cứ thế.

Viết tổng quát thì biểu thức là  $x_n = x_{n-1} + x_{n-2}$ .

Viết chương trình nhập vào số nguyên dương N và in ra màn hình N số Fibonacci đầu tiên.

**For example:**

Input	Result
13	0 1 1 2 3 5 8 13 21 34 55 89 144

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 10**

Not answered

Mark 0.00 out of 10.00

**[Factorial]**

Viết chương trình tính giai thừa của một số nguyên  $n$  cho trước ( $n! = 1 \times 2 \times \dots \times n$ ).

**Đầu vào**

Đầu vào từ bàn phím chứa duy nhất một số nguyên  $n$  ( $0 < n \leq 10$ ).

**Đầu ra**

In ra màn hình  $n!$ .

**For example:**

Input	Result
3	6

**Answer:** (penalty regime: 0 %)

1	
---	--

**Question 11**

Not answered

Mark 0.00 out of 10.00

**[Perfect Number]**

Số hoàn thiện là số nguyên dương, mà tổng các ước nguyên dương nhỏ hơn nó bằng chính nó.

Viết chương trình kiểm tra xem số nguyên dương  $n$  có phải là số hoàn thiện không.

**Đầu vào**

Đầu vào từ bàn phím chứa số nguyên dương  $n$  ( $n \leq 10^7$ ).

**Đầu ra**

In ra màn hình "yes" nếu  $n$  là số hoàn thiện và "no" nếu ngược lại.

**For example:**

Input	Result
6	yes

**Answer:** (penalty regime: 0 %)

1	
---	--

[Back to Course](#)