/\*生理周期

1600012828

薛畅

20160923

\*/

#include<iostream>

using namespace std;

int main()

{

int p, e, i , d;

cin >> p >> e >> i >> d;

p = p + 23; //防止第一次重合

for (p;p<=21252; p = p + 23) //p每次增加23天

{

for (e;e <= p; e = e + 28) //e每次增加28天

{

if (e == p) //e和p重合

{

for (i; i <= p; i = i + 33) //i每次增加33天

{

if (i == p)

{

cout << "Case 1: the next triple peak occurs in " << p - d << " days." << endl; //输出结果

}

}

}

}

}

system("pause");

return 0;

}

/\*生理周期

1600012828

薛畅

20160923

\*/

#include<iostream>

using namespace std;

int main()

{

int p, e, i, d,a;

cin >> p >> e >> i >> d;

for (a = d+1;;a++)

{

if ((a - p) % 23 == 0) //判断p

break;

}

for (a;; a = a + 23)

{

if ((a - e) % 28 == 0) //判断p和e

break;

}

for (a;; a = a + 23 \* 28)

{

if ((a - i) % 33 == 0) //判断p、e和i

{

cout << "Case 1: the next triple peak occurs in " << a - d << " days." << endl;

break;

}

}

system("pause");

return 0;

}

|  |
| --- |
| ProgramError: Case 0. Time out.  Error 1 in case 0.  MoreLines: Too many lines in output file. . ( 125 : 125 )  **Input of in case 0:**  321 152 69 298  103 172 336 151  74 361 146 9  340 68 183 113  115 293 279 229  290 344 122 361  194 350 115 19  269 22 205 94  339 265 255 325  54 202 66 138  281 350 146 208  213 167 56 151  353 167 323 294  125 315 242 61  256 348 256 27  330 336 63 276  57 139 33 34  342 194 355 199  340 62 246 355  157 256 115 131  ...  **Standard Output:**  ...  **Your Output:**  Case 1: the next triple peak occurs in 1058 days.  Case 2: the next triple peak occurs in 19985 days.  Case 3: the next triple peak occurs in 2480 days.  Case 4: the next triple peak occurs in 19639 days.  Case 5: the next triple peak occurs in 15296 days.  Case 6: the next triple peak occurs in 9727 days.  Case 7: the next triple peak occurs in 1647 days.  Case 8: the next triple peak occurs in 11100 days.  Case 9: the next triple peak occurs in 14780 days.  Case 10: the next triple peak occurs in 3228 days.  Case 11: the next triple peak occurs in 13666 days.  Case 12: the next triple peak occurs in 5812 days.  Case 13: the next triple peak occurs in 20621 days.  Case 14: the next triple peak occurs in 478 days.  Case 15: the next triple peak occurs in 9337 days.  Case 16: the next triple peak occurs in 18960 days.  Case 17: the next triple peak occurs in 4025 days.  Case 18: the next triple peak occurs in 7779 days.  Case 19: the next triple peak occurs in 14015 days.  Case 20: the next triple peak occurs in 13481 days.  ... |
| |  | | --- | |  | |

/\*生理周期

1600012828

薛畅

20160923

\*/

#include<iostream>

using namespace std;

int main()

{

int p, e, i, d, a;

for (int n = 1;; n++)

{

cin >> p >> e >> i >> d;

if ((p == -1) && (e == -1) && (i == -1) && (d == -1))

{

break; //停止

}

else

{

for (a = d + 1;; a++)

{

if ((a - p) % 23 == 0) //判断p

break;

}

for (a;; a = a + 23)

{

if ((a - e) % 28 == 0) //判断p和e

break;

}

for (a;; a = a + 23 \* 28)

{

if ((a - i) % 33 == 0) //判断p、e和i

{

cout << "Case " << n << ": the next triple peak occurs in " << a - d << " days." << endl;

break;

}

}

}

}

system("pause");

return 0;

}

埃博拉来袭

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

int day = 2, n, x, y, k, a;

int i = 0;

cin >> n >> x >> y >>k;

a = n;

for (day; day <= y - 1; day++)

{

n = n+n\*x;

}

for (day = y; day <= k; day++)

{

int p = pow(x, i);

n = n - a\*p;

n = n+n\*x;

i = i + 1;

}

cout << n << endl;

system("pause");

return 0;

}

/\*相关月

1600012828

薛畅

20160923

\*/

#include<iostream>

using namespace std;

int main()

{

int n, a, b, c;

cin >> n;

for (int i = 1; i <= n; i++)

{

cin >> a >> b >> c;

if ((b >= 3) && (c >= 3)) //两个月份都大于等于3的情况，不用考虑闰年

{

if (((b == 3) && (c == 11)) || ((c == 3) && (b == 11)) || ((b == 9) && (c == 12)) || ((b == 12) && (c == 9)) || ((b == 4) && (c == 7)) || ((b == 7) && (c == 4)))

{

cout << "YES" << endl; //成相关月的月份为有限个可以直接算出

}

else

{

cout << "NO" << endl;

}

}

else //两个月份中若存在1或2月需考虑闰年的情况

{

if (a % 4 != 0)

{

if (((b == 1) && (c == 10)) || ((b == 10) && (c == 1)) || ((b == 2) && (c == 11)) || ((b == 11) && (c == 2)) || ((b == 2) && (c == 3)) || ((b == 3) && (c == 2)))

{

cout << "YES" << endl; //成相关月的月份为有限个可以直接算出

}

else

{

cout << "NO" << endl;

}

}

else

{

if (a % 100 == 0) //能整除100

{

if (a % 400 == 0) //是闰年

{

if (((b == 1) && (c == 4)) || ((c == 4) && (b == 1)) || ((b == 1) && (c == 7)) || ((b == 7) && (c == 1)) || ((b == 2) && (c == 8)) || ((b == 8) && (c == 2)))

{

cout << "YES" << endl;

}

else

{

cout << "NO" << endl;

}

}

else //不是闰年

{

if (((b = 1) && (c = 10)) || ((b = 10) && (c = 1)) || ((b = 2) && (c = 11)) || ((b = 11) && (c = 2)) || ((b == 2) && (c == 3)) || ((b == 3) && (c == 2)))

{

cout << "YES" << endl;

}

else

{

cout << "NO" << endl;

}

}

}

else

{

if (((b = 1) && (c = 4)) || ((c = 4) && (b = 1)) || ((b = 1) && (c = 7)) || ((b = 7) && (c = 1)) || ((b = 2) && (c = 8)) || ((b = 8) && (c = 2)))

{

cout << "YES" << endl;

}

else

{

cout << "NO" << endl;

}

}

}

}

}

system("pause");

return 0;

}

#include<iostream>

using namespace std;

int main()

{

int sum = 0;

char thisman;

thisman = 'A';

for (int k = 0; k < 4; k = k + 1)

{

thisman = 'A' + k;

sum = (thisman != 'A') + (thisman == 'C') + (thisman == 'D') + (thisman != 'D');

if (sum == 3)

{

break;

}

}

cout << thisman << endl;

system("pause");

return 0;

}

/\*快速找零

1600012828

薛畅

20160926

\*/

#include<iostream>

using namespace std;

int main()

{

int N, prise, Q, D, n, P;

cin >> N;

for (int i = 1; i <= N; i++)

{

cin >> prise; //输入价格

Q = (prise - prise % 25) / 25; //计算Q

D = ((prise - Q \* 25) - (prise - Q \* 25) % 10) / 10; //计算D

n = ((prise - Q \* 25 - D \* 10) - (prise - Q \* 25 - D \* 10) % 5) / 5; //计算n

P = prise - Q \* 25 - D \* 10 - n \* 5; //计算P

cout << i << " " << Q << " QUARTER(S)," << D << " DIME(S)," << n << " NICKEL(S)," << P << " PENNY(S)" << endl;

}

system("pause");

return 0;

}

/\*1600012828

薛畅

20160928

细菌的战争\*/

#include<iostream>

using namespace std;

int main()

{

int n,a,b,k;

cin >> n;

for (int i = 1; i <= n; i++)

{

cin >> a >> b;

if (a <= b)

{

cout << "1" << endl;

}

else

{

for (k = 2;; k++)

{

b = b\*1.05;

a = a - b;

a = a \* 2;

if (a > 100)

{

a = 100;

}

if (a <= 0)

{

break;

}

}

cout << k << endl;

}

}

system("pause");

return 0;

}

/\*1600012828

薛畅

20160928

细菌的战争\*/

#include<iostream>

using namespace std;

int main()

{

int n,a,b,k;

cin >> n;

for (int i = 1; i <= n; i++)

{

cin >> a >> b; //输入初始数量

if (a <= b) //第一次比较

{

cout << "1" << endl;

}

else

{

for (k = 2;; k++)

{

a = a - b; //消灭有害菌

a = a \* 2; //有害菌增长

if (a > 1000000)

{

a = 1000000; //有害菌下降到100万

}

b = b\*1.05; //有益菌增长

if (a <= 0) //有害菌被消灭完

{

break;

}

}

cout << k << endl; //输出天数

}

}

system("pause");

return 0;

}

/\*1600012828

薛畅

20160928

细菌的战争\*/

#include<iostream>

using namespace std;

int main()

{

int n, a, b, k;

cin >> n;

for (int i = 1; i <= n; i++)

{

cin >> a >> b; //输入初始数量

for (k = 1;; k++)

{

a = a - b; //消灭有害菌

a = a \* 2; //有害菌增长

if (a > 1000000)

{

a = 1000000; //有害菌下降到100万

}

b = b\*1.05; //有益菌增长

if (a <= 0) //有害菌被消灭完

{

break;

}

}

cout << k << endl; //输出天数

}

system("pause");

return 0;

}

/\*整数的质因子

1600012828

薛畅

20160928

\*/

#include<iostream>

using namespace std;

int main()

{

int m,sum=0;

cin >> m;

for (int i = 2; i <= m / 2; i++)

{

if (m%i != 0) //该数不是因子

{

continue;

}

else

{

for (int k = 2; k <= i / 2; k++)

{

if (i%k == 0) //该数不是质因子

{

sum = sum + 1;

}

}

if (sum != 0)

{

sum = 0;

continue;

}

else //该数是质因子

{

cout << i << endl;

}

}

}

system("pause");

return 0;

}

/\*密切数

1600012828

薛畅

20160928

\*/

#include<iostream>

using namespace std;

int main()

{

int a, b, c, n, sum=0;

cin >> a >> b;

if (a > b)

{

c = a;

}

else

{

c = b;

}

for (n = 2; n <= c; n++)

{

for (int i = 2; i <= n / 2; i++)

{

if (n%i == 0)

sum = sum + 1;

}

if (sum != 0)

{

continue;

}

else

{

if(a%n==0)&&(b%n==0)

}

}

}

分配病房

int main() { int k, num, flag = 0; //k为病人数，num为病人编号,flag为标记 //threshold为可以住病房的阈值， level为每个病人实际值 double threshold, level; cin >> k >> threshold; for (int i = 0; i < k; i++) { cin >> num >> level; if (level >= threshold) { cout << setw(3) << setfill('0') << num << " " ; cout << fixed << setprecision(1) << level << endl; flag = 1; } } if (!flag) cout << "None" << endl; return 0; }

计算定积分

#include<iostream>

#include<iomanip>

using namespace std;

int main()

{

int a, b, c;

double x1, x2, sum = 0, s, n;

cin >> a >> b >> c >> x1 >> x2;

n = (x2 - x1) / 0.0001;

for (int i = 0; i < n; i++)

{

s = 0.0001\*(a\*(x1 + i\*0.0001)\*(x1 + i\*0.0001) + b\*(x1 + i\*0.0001) + c);

sum = sum + s;

}

cout << fixed;

cout << setprecision(4) << sum << endl;

system("pause");

return 0;

}

密切数判断

#include<iostream>

using namespace std;

int main()

{

int a, b, x, i, sum1 = 0, sum2 = 0;

cin >> a >> b;

x = a > b ? a : b;

for (i = 2; i <= a; i++)

{

for (int n = 2; n <= i / 2; n++)

{

if (i%n == 0)

{

sum1 = sum1 + 1;

}

}

if (sum1 == 0)

{

if ((a%i == 0 && b%i != 0) || (a%i != 0 && b%i == 0))

{

cout << "NO" << endl;

sum2 = sum2 + 1;

break;

}

else

{

sum2 = sum2 + 0;

}

}

}

if (sum2 == 0)

cout << "YES" << endl;

system("pause");

return 0;

}

#include<iostream>

using namespace std;

int main()

{

int a, b, x, i, sum1 = 0, sum2 = 0;

char c;

cin >> a >> c >> b;

x = a > b ? a : b;

if (a != 1 && b != 1)

{

for (i = 2; i <= x; i++)

{

for (int n = 2; n <= i / 2; n++)

{

if (i%n == 0)

{

sum1 = sum1 + 1;

}

}

if (sum1 == 0)

{

if ((a%i == 0 && b%i != 0) || (a%i != 0 && b%i == 0))

{

cout << "NO" << endl;

sum2 = sum2 + 1;

break;

}

else

{

sum2 = sum2 + 0;

}

}

else

{

sum1 = 0;

continue;

}

}

if (sum2 == 0)

cout << "YES" << endl;

}

else

{

if (a == 1 && b == 1)

{

cout << "YES" << endl;

}

else

{

cout << "NO" << endl;

}

}

system("pause");

return 0;

}

#include<iostream>

using namespace std;

int main()

{

char a[101], b[101];

int n;

cin >> n;

for (int i = 1; i <= n; i++)

{

int left=0, right=0;

cin >> a[101];

int len = strlen(a);

for (int j = 0; j < len; j++)

{

if (a[j] == '(')

left += 1;

if (a[j] == ')')

{

if (left == 0)

b[j] = '?';

if (left >= 0)

{

b[j] = ' ';

left -= 1;

}

}

}

for (int j = len - 1; j >= 0; j--)

{

if (a[j] == ')')

right += 1;

if (a[j] == '(')

{

if (right == 0)

b[j] = '$';

if (right >= 0)

{

b[j] = ' ';

right -= 1;

}

}

}

cout << a << endl << b << endl;

}

return 0;

}

回文子串

#include<iostream>

using namespace std;

int main()

{

char a[501];

int i, len, flag = 1;

cin >> a;

len = strlen(a);

for (i = 2; i <= len; i++)

{

for (int j = 0; j <= len-i; j++)

{

for (int k = 0; k <= j + i / 2; k++)

{

if (a[j + k] != a[j + i - k - 1])

{

flag = 0;

break;

}

}

if (flag == 1)

{

for(int k=)

}

}

}

}

/\*两个无符号十进制大整数的减法运算

1600012828

薛畅

20161026

\*/

#include<iostream>

#include<cstring>

using namespace std;

int main()

{

int n, len, num1[201] = { 0 }, num2[201] = { 0 }, x[201] = { 0 }, i, j, y, len1, len2;

char c[401];

cin >> n;

for (int w = 0; w < n; w++)

{

cin >> c; //输入

len = strlen(c);

for (j = 0;; j++)

if (c[j] == '-')

break;

for (int k = j - 1; k >= 0; k--)

num1[200 + k - j] = c[k] - '0';

for (int k = len - 1; k >= j + 1; k--)

num2[200 + k - len] = c[k] - '0';

len1 = j;

len2 = len - j - 1;

if (len1 > len2) //比较长度前一个更大

{

for (int k = 199; k >= 200 - len1; k--)

{

x[k] = num1[k] - num2[k];

if (x[k] < 0)

{

x[k] += 10;

num1[k - 1] -= 1;

}

}

for (i = 0; i <= 199; i++)

if (x[i] != 0)

break;

for (int k = i; k <= 199; k++)

cout << x[k];

cout << endl;

for (int k = 0; k <= 200; k++)

x[k] = 0;

}

if (len2 > len1) //比较长度后一个更大

{

for (int k = 199; k >= 200 - len2; k--)

{

x[k] = num2[k] - num1[k];

if (x[k] < 0)

{

x[k] += 10;

num2[k - 1] -= 1;

}

}

for (i = 0; i <= 199; i++)

if (x[i] != 0)

break;

cout << "-";

for (int k = i; k <= 199; k++)

cout << x[k];

cout << endl;

for (int k = 0; k <= 200; k++)

x[k] = 0;

}

if (len1 == len2) //长度相同

{

for (i = 200 - len1; i <= 199; i++)

if (num1[i] != num2[i]) //继续比较后面的每一项

break;

if (i <= 199)

{

if (num1[i] > num2[i])

{

for (int k = 199; k >= i; k--)

{

x[k] = num1[k] - num2[k];

if (x[k] < 0)

{

x[k] += 10;

num1[k - 1] -= 1;

}

}

for (y = 0; y <= 199; y++)

if (x[y] != 0)

break;

for (int k = y; k <= 199; k++)

cout << x[k];

cout << endl;

for (int k = 0; k <= 200; k++)

x[k] = 0;

}

if (num2[i] > num1[i])

{

for (int k = 199; k >= i; k--)

{

x[k] = num2[k] - num1[k];

if (x[k] < 0)

{

x[k] += 10;

num2[k - 1] -= 1;

}

}

for (y = 0; y <= 199; y++)

if (x[y] != 0)

break;

cout << "-";

for (int k = y; k <= 199; k++)

cout << x[k];

cout << endl;

for (int k = 0; k <= 200; k++)

x[k] = 0;

}

}

if (i == 200)

cout << "0" << endl;

}

}

return 0;

}

/\*K进制数的子序列

1600012828

薛畅

20161026

\*/

#include<iostream>

#include<cstring>

using namespace std;

int main()

{

int k, m, x[31], len, i, j, l, y;

char num[31], a, b;

cin >> k >> a >> m >> b >> num;

len = strlen(num);

for (i = 0; i <= len - 1; i++)

{

x[i] = num[i] - '0';

}

for (j = 0; j <= m / 5; j++)

{

for (l = 1; l <= 5; l++)

{

x[len - 1] = x[len - 1] + 1;

for (y = len - 1; y >= 0; y--)

{

if (x[y] >= k) //向前进一位

{

x[y - 1] += x[y] / k;

x[y] = x[y] % k;

}

else

break;

}

if (l == 1) //每一行的第一个数字

for (i = 0; i <= len - 1; i++)

cout << x[i];

else

{

cout << ",";

for (i = 0; i <= len - 1; i++)

cout << x[i];

}

if (l == 5) //最后一个换行

cout << endl;

if (5 \* j + l == m)

break;

}

if (5 \* j + l + 1== m) //判断是否结束

break;

}

return 0;

}

/\*K进制数的子序列

1600012828

薛畅

20161026

\*/

#include<iostream>

#include<cstring>

using namespace std;

int main()

{

int k, m, x[31], len, i, j, l, y;

char num[31], a, b;

cin >> k >> a >> m >> b >> num;

len = strlen(num);

for (i = 0; i <= len - 1; i++)

{

x[i] = num[i] - '0';

}

for (j = 0; j <= m / 5; j++)

{

for (l = 1; l <= 5; l++)

{

x[len - 1] = x[len - 1] + 1;

for (y = len - 1; y >= 0; y--)

{

if (x[y] >= k) //向前进一位

{

x[y - 1] += x[y] / k;

x[y] = x[y] % k;

}

else

break;

}

if (l == 1) //每一行的第一个数字

for (i = 0; i <= len - 1; i++)

cout << x[i];

else

{

cout << ",";

for (i = 0; i <= len - 1; i++)

cout << x[i];

}

if (l == 5) //最后一个换行

cout << endl;

if (5 \* j + l == m)

break;

}

if (5 \* j + l - 1== m) //判断是否结束

break;

}

return 0;

}