```c
/* http://www.ioccc.org/years.html */

#include "stdio.h"
#define e 3
#define g (e/e)
#define h ((g+e)/2)
#define f (e-g-h)
#define j (e*e-g)
#define k (j-h)
#define l(x) tab2[x]/h
#define m(n,a) ((n&(a))==(a))

long tab1[]={ 989L,5L,26L,0L,88319L,123L,0L,9367L };
int tab2[]={ 4,6,10,14,22,26,34,38,46,58,62,74,82,86 };

main(m1,s) char *s; {
    int a,b,c,d,o[k],n=(int)s;
    if(m1==1){ char b[2*j+f-g]; main(l(h+e)+h+e,b); printf(b); }
    else switch(m1-=h){
    case f:
        a=(b=(c=(d=g)<<g)<<g)<<g;
        return(m(n,a|c)|m(n,b)|m(n,a|d)|m(n,c|d));
    case h:
        for(a=f;a<j;++a)if(tab1[a]&&!(tab1[a]%((long)l(n))))return(a);
    case g:
        if(n<h)return(g);
        if(n<j){n-=g;c='D';o[f]=h;o[g]=f;}
        else{c='\r'-'\b';n-=j-g;o[f]=o[g]=g;}
        if((b=n)>=e)for(b=g<<g;b<n;++b)o[b]=o[b-h]+o[b-g]+c;
        return(o[b-g]%n+k-h);
    default:
        if(m1-=e) main(m1-g+e+h,s+g); else *(s+g)=f;
        for(*s=a=f;a<e;) *s=(*s<<e)|main(h+a++,(char *)m1);
    }
}

```

```c
/*****************************************************************************
 * math1.c
 *
 * David J. Malan
 * malan@harvard.edu
 *
 * Computes a total but does nothing with it.
 *
 * Demonstrates use of variables.
 *****************************************************************************/

#include <stdio.h>

int main(void)
{
    int x = 1;
    int y = 2;
    int z = x + y;
    return 0;
}
```

```c
/****************************************************************************
 * math2.c
 *
 * David J. Malan
 * malan@harvard.edu
 *
 * Computes and prints an integral total.
 *
 * Demonstrates use of a format string.
 ****************************************************************************/

#include <stdio.h>

int main(void)
{
    int x = 1;
    int y = 2;
    int z = x + y;
    printf("%d\n", z);
    return 0;
}
```

```c
/*****************************************************************************
 * math3.c
 *
 * David J. Malan
 * malan@harvard.edu
 *
 * Computes and prints a floating-point total.
 *
 * Demonstrates loss of precision.
 *****************************************************************************/

#include <stdio.h>

int main(void)
{
    float answer = 1 / 10;
    printf("%.2f\n", answer);
    return 0;
}
```

```c
/*****************************************************************************
 * math4.c
 *
 * David J. Malan
 * malan@harvard.edu
 *
 * Computes and prints a floating-point total.
 *
 * Demonstrates use of floating-point math.
 *****************************************************************************/

#include <stdio.h>

int main(void)
{
    float answer = 1 / 10.0;
    printf("%.2f\n", answer);
    return 0;
}
```