

Re: please check my homework

Source: <http://coding.derkeiler.com/Archive/General/comp.programming/2005-10/msg00686.html>

- *From:* "Michael Jørgensen" <ccc59035@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 26 Oct 2005 08:39:24 +0200
-

"David Marsh" <dmarsh@xxxxxxxx> wrote in message
[news:sAw7f.306880\\$t12.244927@xxxxxxxxxxxx](mailto:news:sAw7f.306880$t12.244927@xxxxxxxxxxxx)

> This is a homework assignment for extra credit. The statistical
> material has not been covered in class. Is my bidist() function
> correct? It seems to work with the examples I have tried, but I
> would appreciate your comments and suggestions. PLEASE don't
> write any code. Error checking deliberately omitted for now and I
> am aware of the potential for factorials to exceed LDBL_MAX.

>

> David

> -----

>

> #include <stdio.h>

> #include <stdlib.h>

> #include <math.h>

>

> long double bidist(long double p, long double n, long double x);

> long double factorial(long double n);

>

> int main(int argc, char *argv[])

> {

> long double pp, nn, xx, result;

>

> if(argc != 4) exit(EXIT_FAILURE);

>

> pp = strtold(argv[1], NULL);

> nn = strtold(argv[2], NULL);

> xx = strtold(argv[3], NULL);

>

> result = bidist(pp, nn, xx);

> printf("%Lf\n", result);

>

> return 0;

> }

>

> /* return the probability of x successes in n events,

> given the probability p for success in a single event,

> using the formula for the binomial distribution:

Re: please check my homework

```
> n!  
> f(x) = ----- ( p^x )( (1-p)^(n-x) )  
> x! (n-x)!  
> */
```

In the above comment you should specify that n must be a positive integer, that x must be an integer in the interval [0..n], and that p must be a real number in the interval [0..1].

Write also a test function that calculates: $f(0) + f(1) + f(2) + \dots + f(n-1) + f(n)$. This sum should be exactly 1.0 for all valid values of n and p.

Also, make an estimate of how large values of n your program allows, without running into the problem with `LDBL_MAX`. For instance $100! = 9.33 \cdot 10^{257}$.

I'm very concerned with the problem of overflow. You've already recognized the problem, but you should at least consider what alternatives there are. How could you deal with the problem, if it became necessary?

Performance might be a problem too. You have three calls to factorial, each involving a loop. For large values of n and x this might be very slow. If necessary, how would you deal with that?

```
> long double bidist(long double p, long double n, long double x)  
> {  
> long double fn, fx, fnx, t1, t2, t3;  
>  
> fn = factorial(n);  
> fx = factorial(x);  
> fnx = factorial(n-x);  
> t1 = fn / (fx * fnx);  
> t2 = (long double)pow(p, x);  
> t3 = (long double)pow(1.0-p, n-x);  
>  
> return t1 * t2 * t3;  
> }  
>  
> /* return n! */  
> long double factorial(long double n)  
> {  
> long double f;  
>  
> f = 1.0;  
> while(n > 1.0)  
> f *= n--;  
>  
> return f;  
> }
```

-Michael.

Re: please check my homework

- *Follow-Ups:*
 - ◆ *Re: please check my homework*
 - ◇ *From:* David Marsh
 - ◆ *Re: please check my homework*
 - ◇ *From:* David Marsh

- *References:*
 - ◆ *please check my homework*
 - ◇ *From:* David Marsh

- Prev by Date: *Re: Struct vs Union on TCP/IP code*
- Next by Date: *Re: Cellular Automa*
- Previous by thread: *please check my homework*
- Next by thread: *Re: please check my homework*
- Index(es):
 - ◆ *Date*
 - ◆ *Thread*