

Re: [C][OT] erroneous fprintf output

Source: <http://coding.derkeiler.com/Archive/C/ CPP/alt.comp.lang.learn.c-cpp/2004-05/0504.html>

From: Curley Q. (*curleyq_at_bogus.net*)

Date: 05/14/04

Date: Thu, 13 May 2004 22:07:34 GMT

Francis Glassborow wrote:

> *My criticism is that awful selection mechanism that isn't even*
> *guaranteed to ever terminate when creating a single shuffle of the 100*
> *numbers. If you want to do it by random selection use a standard shuffle*
> *function:*
>
> `void swap_random_with_last(int array[], int size){`
> `int choice = rand() % size;`
> `int temp = array[choice];`
> `array[choice] = array[size-1];`
> `array[size-1] = temp;`
> `}`
>
> `void shuffle(int array[], int size){`
> `int i;`
> `for(i = size; i != 1; i--){`
> `swap_random_with_last(array, i);`
> `}`
>
> *Now create an array of 100 entries and initialise them with 1 to 100 and*
> *call shuffle.*
>
> *However are you required to have an equal number of items in each (i.e.*
> *50 : 50)?*

No

> *Consider creating two queues (stacks or whatever other image you like)*
> *and add the next square-root to whichever stack currently has the*
> *smaller total. Initially the values will go on alternating piles, but*
> *eventually the one that is getting the even numbers will get far enough*
> *ahead so that the square roots of a consecutive pair will both get added*
> *to the first pile, then they will alternate again. Actually*
> *investigating the pattern of when successive square roots go onto the*
> *same pile could be quite interesting, but that is another problem.*
>
> *If you are dealing with 100 numbers you just might find that this*

> *algorithm results in a pile of 51 and another of 49 but no worse than*
> *that (proving that is not very hard).*

Have I correctly implemented your suggestions in the code below?

```
/* calculate the sqrts of 1-100 and divide
   them into 2 sets that are approx equal */
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define SIZE 100
void swap_random_with_last(int array[], int size);
void shuffle(int array[], int size);

int main(void)
{
    double sqroots[SIZE];
    int integers[SIZE];
    double set1, set2, total;
    int idx;

    set1 = set2 = total = 0.0;

    for(idx = 0; idx < SIZE; idx++)
        integers[idx] = idx + 1;

    srand((unsigned int)time(0));

    shuffle(integers, SIZE);

    for(idx = 0; idx < SIZE; ++idx)
    {
        sqroots[idx] = sqrt((double)integers[idx]);
        total += sqroots[idx];
    }

    for(idx = 0; idx < SIZE; ++idx)
        if(set1 == set2 || set1 < set2) /* equal on 1st pass */
            set1 += sqroots[idx];
        else
            set2 += sqroots[idx];

    printf("set1 = %f\n", set1);
    printf("set2 = %f\n", set2);
    printf("total = %f\n", total);

    return 0;
}

void swap_random_with_last(int array[], int size){
    int choice = rand() % size;
```

alt.comp.lang.learn.c-c++: Re: [C][OT] erroneous fprintf output

```
int temp = array[choice];
array[choice] = array[size-1];
array[size-1] = temp;
}
```

```
void shuffle(int array[], int size){
    int i;
    for(i = size; i != 1; i--)
        swap_random_with_last(array, i);
}
```

> *When you have two piles you may need to fine tune to get them more equal*

To say the least.