

Re: weird problem

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2007-01/msg04061.html

- *From:* jt@xxxxxxxxxxxx (Jens Thoms Toerring)
 - *Date:* 26 Jan 2007 23:57:04 GMT
-

Alef.Veld@xxxxxxxx wrote:

Hmm, i thought i had replied to this. Probably clicked on discard. Ok, one more time:

In order to make clearer what's going wrong in your code I have cut it down to a one-dimensional case, removed a few non-essential things and removed all the error checking (which is definitely necessary but gets a bit in the way here). I also tried to re-indent it and put in some spaces e.g. around operators etc. for easier readability. After doing that you end up with something like the following. I first list it as it is and then again with my comment. Since it's not a direct copy of your post I use a '|' instead of the of the usual '>' to indicate what basically came from you.

```
| GLvoid display_grid(GLvoid)
| {
| objects *tmp = o_head;
|
| while (tmp != NULL) {
| free(tmp->x);
| assign_coordinate(tmp);
| tmp = tmp->next;
| }
| }
|
| GLvoid assign_coordinate(objects *tmp)
| {
| objects *search = o_head;
|
| tmp->x = malloc(sizeof(float) + 7);
| snprintf(o_tail->x, sizeof(float) + 7, "%.0f",
| ((grid_size - grid_start) * rand()) / RAND_MAX + grid_start);
|
| while (search != NULL) {
|
| if (collisions>0 &&
| collisions ==((grid_size - grid_start) * (grid_size -
```

Re: weird problem

```
| grid_start)))  
| exit(1);  
|  
| if(tmp != search &&  
| !strcmp(tmp->x, search->x)) {  
| collisions++;  
| free(tmp->x);  
| assign_coordinate(tmp);  
| }  
|  
| search=search->next;  
| }  
| }
```

I hope I didn't miss anything essential. Now again, this time with comments.

```
| GLvoid display_grid(GLvoid)  
| {  
| objects *tmp = o_head;  
|  
| while (tmp != NULL) {
```

So here you loop over all the elements of a linked list, starting with 'o_head'. Fine so far.

```
| free(tmp->x);  
| assign_coordinate(tmp);  
| tmp = tmp->next;  
| }  
| }  
|  
| GLvoid assign_coordinate(objects *tmp)  
| {  
| objects *search = o_head;  
|  
| tmp->x = malloc(sizeof(float) + 7);
```

Here you allocate memory for the 'tmp->x' member you just before you called assign_coordinate() had free()ed. Fine so far in that respect. The question of your use of 'sizeof(float) + 7' I am going to address at the end.

```
| snprintf(o_tail->x, sizeof(float) + 7, "%.0f",  
| ((grid_size - grid_start) * rand()) / RAND_MAX + grid_start);
```

Now you print some text into 'o_tail->x' – please look again at what you are doing here. This isn't 'tmp->x' (as you told in a different post, 'o_tail' is the last element of the linked list and that's not what 'tmp->x' is except possibly in a single case). It seems at least

Re: weird problem

strange that you put some random value into the last element of the linked list and don't use the newly allocated memory, pointed to by 'tmp->x' for that purpose, especially considering what's happening later.

```
| while (search != NULL) {  
|  
| if (collisions>0 &&  
| collisions ==((grid_size - grid_start) * (grid_size -  
| grid_start)))
```

I already told you that the comparison between an integer and a float value for equality is rather likely not to work – you have to round the result of the floating point calculation to the nearest integer before you can do a meaningful comparison. Moreover, since you seem to want to check if there are as many collisions as there are points in your grid you make an off-by-one error. Just draw a simple grid on paper and count the number of places where the lines intersect...

```
| exit(1);  
|  
| if(tmp != search &&  
| !strcmp(tmp->x, search->x)) {
```

And here things go completely wrong: you never ever set the memory 'tmp->x' points to anything, so 'tmp->x' points to uninitialized memory. So it is definitely not suitable to be passed as an argument to strcmp() which expects a pointer to a string. My only guess is that you actually intend to put the random float value you produced above into 'tmp->x' instead of 'o_tail->>x', in which case the strcmp() here might make some sense, especially since you seem to want to check if the random value hasn't been used somewhere else in your list.

And now a question about something else: why do you use floating point numbers for your grid points when you obviously only need integers? You always round them anyway (by using the "%.0f" conversion specifier), so why use floats at all? Lots of things are probably going to be quite a bit simpler if you use integers instead.

```
| collisions++;  
| free(tmp->x);  
| assign_coordinate(tmp);  
| }  
|  
| search=search->next;  
| }  
| }
```

And now concerning the 'sizeof(float) + 7' thing from above (we al-

Re: weird problem

Re: weird problem

first guess, allocates more memory accordingly and then tries again. Repeat until you had success. Those are the only options you have as far as I can see at the moment.

Regards, Jens

PS: Please be so kind to follow the convention normally used in technical newsgroups and don't top-post. Top-posting means putting your text before what you are replying to. This is rather annoying since it makes it hard to figure out what you are reacting to. So please try to put your replies below a citation of what you are reacting to, but make sure you only cite what is still relevant to your reply. That way the readers get an idea what's it all about without having to sift through long and partly irrelevant texts. This, in turn, will increase the likelihood of getting useful replies to your questions.

--

\Jens Thoms Toerring ___ jt@xxxxxxxxxxx

_____ <http://toerring.de>

.