

Ruby - Misc #13230

Better Do ... while structure

02/19/2017 01:29 AM - jzakiya (Jabari Zakiya)

Status:	Rejected	
Priority:	Normal	
Assignee:		
Description		
I just saw this, and thought I'd pass it along.		
http://ncomputers.org/suggestions/do%20while.cpp		
Do ... while structure improvement Sometimes the use of jumps such as: continue, break, goto, call to a function, etc. is necessary to avoid the execution of some instructions.		
This is the case of the seed, warp and swap loops of this solution for the n queens problem.		
To avoid the use of jumps or tricks like for(;;) if(condition)break; on some of these cases, we are suggesting the below improvement to the do ... while loop str		
<pre>/* author: ncomputers.org */ int main(){ // Allow initializers (extra improvement) do(bool condition=0){ // Block A }while(condition){ // Block B // Variables declared inside the initializer still visible } return 0; }</pre>		

History

#1 - 02/20/2017 05:11 AM - shyouhei (Shyouhei Urabe)

- Status changed from Open to Feedback

I'm sorry but I have to say I don't understand how the construct works. Looking at the original site there is an example:

```
/* author: ncomputers.org */
#include<iostream>
using namespace std;

int main(){
    unsigned a[]={0,1,2,3,4,5,6,7,8,9};
    do(unsigned b=0){
        cout<<a[b];
    }while(++b<10){
        cout<<', ';
    }
    cout<<endl;
    return 0;
}
```

1. At sight, I have no idea what this code does, nor how it works.
2. By closely looking at the description in <http://ncomputers.org/examples/do%20while%201.cpp>, it says "prints the content of an array, separating each of its elements by a comma". Now I'm totally puzzled. I don't understand why that is possible by the code, especially how the variable b changes over time.

Maybe I'm just a hard-headed nuts.

#2 - 02/20/2017 05:35 AM - nobu (Nobuyoshi Nakada)

Probably it equals:

```
unsigned b=0;
do {
    cout<<a[b];
    if (!(++b<10)) break;
    cout<<',';
} while (1);
```

or

```
for (unsigned b=0; cout<<a[b], (++b<10);) {
    cout<<',';
}
```

If it were in Ruby:

```
b = 0
while true
  STDOUT << a[b]
  break unless (b += 1) < 10
  STDOUT << ','
end

b = 0
while (STDOUT << a[b]; (b += 1) < 10)
  STDOUT << ','
end

b = 0
while begin
  STDOUT << a[b]
  (b += 1) < 10
end
  STDOUT << ','
end
```

#3 - 04/17/2017 06:53 AM - matz (Yukihiro Matsumoto)

- *Status changed from Feedback to Rejected*

You haven't explained the reason behind the proposal. We don't understand why you think it's "better". If you think you can persuade us with real-world use-case, try to reopen the issue.

Matz.