
Subject: Re: Some new functions

Posted by [dolik.rce](#) on Sat, 20 Nov 2010 19:55:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Fun fact: I tried to come up with optimal solution for Even() and all of the following appear to have same speed as the one from Koldo (with gcc optimal+speed flag):
`inline bool even1(int val) {return !(val&1);}`

`inline bool even2(int val) {return ~val&1;}`

`inline bool even3(int val) {return !(val%2);}`

Without the speed flag even2() seems to be slightly faster.

Also `val%2` and `val&1` for Odd() yields the same speed in both cases

However, the proposed RoundEven() function is suboptimal thanks to the branching. Even though it won't probably be used often, I would suggest faster version:
`inline int roundeven(int val) {return ((1+val)>>1)<<1;}`

//for completeness also rounding to odd numbers:

`inline int roundodd(int val) {return ((val>>1)<<1)+1;}`

Regarding the image access: The `img[y][x]` is great, but still it would be nice to have a wrapper that would allow to put the arguments in (imho) more natural order. For example something like
`RGBA* Image::Get(int x,int y){return (*this)[y][x];}`

Best regards,

Honza
