Subject: Bug with FindMatch(const Range& r, const C& match, int from = 0)
Posted by Novo on Tue, 06 Feb 2018 04:00:04 GMT
View Forum Message <> Reply to Message

This method should look like below.

```
template <class Range, class C>
int FindMatch(const Range& r, const C& match, int from = 0)
{
 for(int i = from; i < r.GetCount(); i++)
  if(match(r[i])) return i;
 return -1;
}
```

Template parameter class V is redundant. It doesn't allow to deduct types from arguments.

Subject: Re: Bug with FindMatch(const Range& r, const C& match, int from = 0)
Posted by mirek on Tue, 06 Feb 2018 08:37:25 GMT
View Forum Message <> Reply to Message

Thanks, fixed.

Anyway, I am intrigued why this was not causing problems? FindMatch is used in FindIndex and that one is used quite intensely, so e.g. theide should not compile...

What compiler are you using?

Mirek

Subject: Re: Bug with FindMatch(const Range& r, const C& match, int from = 0)
Posted by Novo on Tue, 06 Feb 2018 15:31:09 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 06 February 2018 03:37Thanks, fixed.

Anyway, I am intrigued why this was not causing problems? FindMatch is used in FindIndex and that one is used quite intensely, so e.g. theide should not compile...

What compiler are you using?

Mirek
gcc 7.2.0

FindIndex looks like below.

```
template <class Range, class V>
int FindIndex(const Range& r, const V& value, int from = 0)
```

```
{
 for(int i = from; i < r.GetCount(); i++)
  if(r[i] == value) return i;
 return -1;
}
```

FindMatch is not used there.
Actually, I couldn't find even a single use of FindMatch in Upp ...

Subject: Re: Bug with FindMatch(const Range& r, const C& match, int from = 0)
Posted by mirek on Tue, 06 Feb 2018 16:17:19 GMT
View Forum Message <> Reply to Message

I see, correct. I have mistaken it with CoFindMatch....
