Subject: Re: get_i
Posted by mirek on Tue, 16 Jun 2020 19:20:19 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 16 June 2020 18:21Another experiment/suggestion.
I rewrote get_i using variadic template:

```
template <typename A, typename... T>
constexpr A get_i(int i, const A& p0, const T& ...args)
{
 A list[] = {p0, args...};
 int n = sizeof...(args);
 return list[clamp(i, 0, n)];
}

const char* cr = get_i(1, "zero", "one", "two");
RDUMP(cr);
int ir = get_i(1, 0, 1, 2);
RDUMP(ir);
```

IMHO, my implementation is much shorter and it will compile faster.
IMHO, macroses __List and __Expand are not needed anymore ...

Yes, you are right about this, I have used old tricks mostly out of habit. I guess I will have to rewrite it all now :)

However, constexpr I still do not agree. Following your logic, we should add constexpr to every single function everywhere - these are as likely to have constant parameters as get_i (which has like 0.00000001% chance that first parameter will be const in real code).