

Korisnički definisani literali

- ❖ Korisnički definisani literali imaju isti format kao i ugrađeni literali, s tim što imaju sufiks koji mora da počne donjom crtom `_` i malim slovom iza nje; sami literali mogu biti u formatu celobrojnih, racionalnih, znakovnih ili string-literala, ali bez sufiksa koji se koriste za takve ugrađene literale
- ❖ Literalski operatori moraju da imaju naziv *operator*"" , i za koga sledi identifikator koji počinje donjom crtom i malim slovom iza nje. Kao parametre mogu da imaju samo neke od dozvoljenih tipova, u koje spadaju samo *unsigned long long* za celobrojne literale, *long double* za racionalne literale, *char* i njegove varijacije za znakovne literale i *const char** i varijacije za string literale (za njih se koristi i drugi argument koji daje veličinu niza znakova u string-literalu). Na primer:

```
constexpr long double operator"" _deg (long double deg) {  
    return deg*3.1415926/180;  
}
```

- ❖ Korisnički string-literali mogu da se koriste za identifikaciju instanci apstrakcije koje se ne mogu predstaviti numeričkim literalima; na primer:

```
class Date {  
public:  
    Date (unsigned y, unsigned m, unsigned d);  
    ...  
};  
  
Date operator"" _date (const char*, size_t);  
  
Date today = "04.11.2018." _date;
```

Specifikator *constexpr* navodi da se entitet u deklaraciji može izračunati za vreme prevođenja i koristiti u konstantnim izrazima. Pošto je argument ove funkcije literal (vrednost poznata za vreme prevođenja), i pošto je celo telo funkcije izračunljivo za vreme prevođenja, sve može biti konstantan izraz

Poziva se *operator"" _date*("04.11.2018.", 11)

Glava 8: Deklaracije i opseg važenja

- ❖ Deklaracije
- ❖ Opseg važenja
- ❖ Opseg važenja bloka
- ❖ Opseg važenja klase
- ❖ Prostori imena
- ❖ Enumeracije sa opsegom važenja

