

Pretprocesiranje

- ❖ Direktivom *#define* uvodi se nov simbol (identifikator) u pretprocesiranje (u tabelu simbola pretprocesora definisanih ovom direktivom)
- ❖ Opciono se može definisati i *makrozamena*: svaka pojava datog simbola zamenjuje se datim tekstom u nastavku direktive; mogući su i argumenti makrozamene:

```
#define _a_h
```

Jednostavno definiše simbol `_a_h` u tabelu simbola pretprocesora

```
#define N 10
```

Svaka pojava simbola `N` zamenjuje se tekstom (literalom 10)

```
#define max(a,b) (((a)>=(b))?(a):(b))
```

Makrozamena sa argumentima

```
int a[N];
```

Prevodilac će videti deklaraciju: `int a[10];`

```
void f (int a, int b){
```

```
...
```

```
int c = max(x/3,y+1);
```

```
...
```

Prevodilac će videti deklaraciju:

```
int c = (((x/3)>=(y+1))?(x/3):(y+1));
```

```
}
```

Pretprocesiranje

- ❖ Direktive `#if`, `#elif` (znači “*else-if*”), `#ifdef` (znači “*if-defined*”) i `#ifndef` (znači “*if-not-defined*”) omogućuju uslovno prevođenje: tekst uokviren odgovarajućim direktivama biće preveden samo ako je odgovarajući uslov ispunjen; za direktive `#ifdef`/`#ifndef`, uslov je ispunjen ako je dati simbol definisan/nije definisan pretprocesoru (pomoću prethodne direktive `#define`):

```
#ifdef _a_h
```

```
...
```

```
#endif
```

- ❖ Ove direktive mogu da se koriste npr. za prevođenje delova koda koji treba da postoji samo ako se program prevodi za odgovarajuću platformu, konfiguraciju i slično; prevodioci dozvoljavaju da im se zada simbol pretprocesora koji se smatra generalno (globalno) definisanim za sve fajlove, iako nije eksplicitno definisan direktivom `#define`:

```
#ifdef Windows
```

```
... // Windows-dependent code
```

```
#elif defined(Linux)
```

```
... // Linux-dependent code
```

```
#endif
```