

funkce main

- Musí být v programu přítomna (u některých překladačů má upravené jméno) je to funkce, které překladač předá řízení
- Funkce main musí vracet celočíselnou hodnotu.
- V seznamu parametrů jsou jí předána data z programové řádky při volání (tato část může zůstat prázdná)

Zkrácený tvar

```
int main( )    // hlavička funkce: návratový typ, jméno funkce, seznam parametrů
{              // začátek programového bloku – tělo funkce musí být blok
    return 0;   // zapsání návratové hodnoty funkce pro další použití
}              // konec bloku – těla funkce
```

Plný tvar

```
int main(int argc, char *argv[], char *env[])
{ return 0; }
```

nejčastější tvar

```
int main(int argc, char *argv[])
```

založte v programovém prostředí projekt, v něm vytvořte soubor s01_main.c, a napište funkci main.

hlavičku main upravte tak, aby program pracoval s parametry programové řádky (například dostane dva názvy souborů – z prvního přkopíruje data do druhého)

Při volání/spuštění programu je často nutné mu předat informace k upřesnění jeho činnosti. Například při kopírování souborů je nutné (aby byl program univerzálně použitelný) programu říci jak se jmenují zdrojový a cílový soubor (odkud kam se kopíruje).

volání programu na příkazové řádce je potom `c:>kopiruj.exe zdroj.txt cil.txt`

Jak se program dozví o těchto parametrech? K předání slouží parametry funkce `main`

při prototypu funkce `main`: `int main(int argc, char *argv[])`

bude v `argc` (argumentů count=počet argumentů) hodnota 3, protože na příkazovém řádku jsou tři řetězce (oddělené mezerami)

Konkrétní řetězce budou v proměnné `argv` (argumenty value = hodnoty argumentů) tedy:

`argv[1]` bude “kopiruj.exe“

`argv[2]` bude “zdroj.txt“

`argv[3]` bude “cil.txt“

co definuje a jak čteme definici: `char *argv[]`. Jak je uloženo v paměti?

program může vypadat například takto

```
int main(int argc, char *argv[])
{
FILE *ur, *uw;
if (argc != 3)
{
printf("chybny pocet argumentu – chci tri."); return 1;
}

if ((ur = fopen(argv[1], "rt")) == NULL)
{
printf("nepodarilo se otevrit vstupni soubor %s", argv[1]); return 2;
}
```