

## Dyrektywy preprocesora, funkcja `main`

## Struktura programu

dyrektywy preprocesora

deklaracje i definicje globalnych danych i struktur danych

deklaracje i definicje funkcji

definicja funkcji `main`

## Preprocesor

# dyrektywa      argumenty

Dołączanie plików bibliotecznych :      `#include`

`#include`      `<stdio.h>`

`#include`      `"funkcje.c"`

## Włączanie plików do kompilacji

zawartość pliku o nazwie 'plik'

```
printf("Ala ma kota\n");  
}
```

---

```
#include "stdio.h"
```

```
void print(){  
#include "plik"
```

```
int main() {  
    print();  
    return 0;  
};
```

## Zastępowanie tekstów

```
#define ROZMIAR 150

#define moje cena_zakupu - cena_sprzedazy

#define EPS 3.5E-8

.....

#undef EPS // usunięcie definicji

#define EPS 1.5E-8
```

## Ujęcie w nawiasy

```
#define TEMPERATURE_WITH (-1)

#define TEMPERATURE_WITHOUT -1

// ćwiczenie do zweryfikowania

int w = 10-TEMPERATURE_WITH;

int wo = 10-TEMPERATURE_WITHOUT;
// czy kompiluje się?
```

## Zastępowanie tekstów (poprawka)

```
#define ROZMIAR (150)

#define moje ((cena_zakupu) - (cena_sprzedazy))

#define EPS (3.5E-8)

.....

#undef EPS // usunięcie definicji

#define EPS (1.5E-8)
```

## Makra

```
#define Makro1(x) (sin(x) + 3 * (x));
.....
double akr = 2.544;
Makro1(akr) // (sin(akr) + 3 * (akr));

#define Makro2(x, y) ((x) + (y) - 1);
.....
double alfa = -12.74, beta = 0.21;
Makro2(alfa, beta) // ((alfa) + (beta) - 1);
```

## Ujęcie w nawiasy c.d.

```
// Aby utrzymać kolejność wykonywania działań
#define quad(x) (x*x)

int a = quad(2+3);

// zostanie rozwinięte do
int a = (2+3*2+3);

// powinno być
#define quad(x) ((x)*(x))
```

---

---

---

---

---

---

## Ujęcie w nawiasy c.d.

```
// niebezpieczne!!!
// jaki będzie wynik?

#define quad(x) ((x) * (x))

int i = 1;

int a = quad(i++);

// zostanie rozwinięte do
int a = (i++) * (i++);

warning: operation on 'i' may be undefined
```

---

---

---

---

---

---

## Kompilacja warunkowa

```
#if   wyrażenie Stałe_1
    tekst_źródłowy_1
#elif wyrażenie Stałe_2
    tekst_źródłowy_2
.....
#else
    tekst_źródłowy_n
#endif
```

---

---

---

---

---

---

## Test zdefiniowania

```
#defined identyfikator
// 1 : gdy identyfikator był już zdefiniowany
// 0 : gdy identyfikator jest nie zdefiniowany

#if defined identyfikator
/* równoważne */
#ifdef identyfikator

#if ! defined identyfikator
/* równoważne */
#ifndef identyfikator
```

---

---

---

---

---

---

```
#define WersjaProbna
```

```
//
```

```
#ifdef WersjaProbna
```

```
.....
```

```
#else
```

```
.....
```

```
#endif
```

## Polecenia dla kompilatora (poza standardem)

```
#pragma message "WersjaProbna"
```

```
// podczas kompilacji
```

```
#pragma once // jednokrotne dołączenie pliku
```

## Funkcja main

```
void main( void )  
{ ... }
```

```
void main ( )  
{ ... }
```

```
int main ( )  
{  
    .....  
    return 0;  
}
```

## Parametry wejściowe

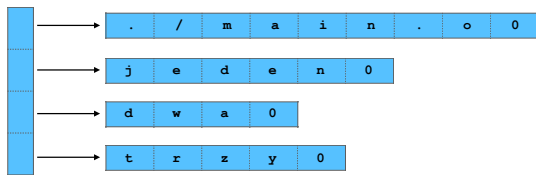
```
void main ( int liczba_slow,  
            char *tabela_slow [ ] )
```

```
{ ... }
```

./main.o jeden dwa trzy

liczba\_słów = 4

tabela\_słów



---

---

---

---

---

---