

Zawartość plików nagłówkowych (*.h) :

<code>#include <stdio.h></code>	dyrektywy dołączenia
<code>#define ESC 27</code>	definicje stałych
<code>#define MAX(x,y) ((x)>(y)?(x):(y))</code>	definicje makr
<code>int menu(char* tab[], int ilosc);</code>	prototypy funkcji
<code>struct TOsoba { char nazwisko[30]; int wiek; };</code>	definicje typów
<code>inline double Kwadrat(double x) { return x*x; }</code>	definicje funkcji „inline”
<code>class TSpisOsob;</code>	deklaracje nazw klas i struktur
<code>extern int licznik;</code>	deklaracje zmiennych
<code>template <class Element> class Tablica { Element t[10]; }</code>	szablony klas i funkcji

Co nie powinno być umieszczane w plikach nagłówkowych:

<code>int x,y; double z; long tablica[10];</code>	nie umieszczamy definicji zmiennych (ale można umieścić deklaracje np <code>extern long tablica[10];</code>)
<code>int Suma(int a,int b) {return a+b; }</code>	nie umieszczamy definicji zwykłych funkcji

Podstawowe dyrektywy (polecenia) preprocesora:

- 1) Dyrektywa **#include <stdio.h>**
oznacza dołączenie w miejscu wystąpienia tego polecenia „standardowego” pliku nagłówkowego,
dyrektywa **#include "tablica1.h"**
oznacza dołączenie pliku nagłówkowego „użytkownika”
- 2) Dyrektywa **#define nazwa** tworzy makrodefinicję,
#undef nazwa unieważnia makrodefinicję
- 3) Polecenia kompilacji warunkowej pozwalają na kompilację tylko jednej z sekcji instrukcji:

```
#if wyrażenie1  
    sekcja_instrukcji1  
#elif wyrażenie2  
    sekcja_instrukcji2  
#else  
    końcowa_sekcja_instrukcji  
#endif
```

- 4) Polecenia warunkowej kompilacji uniemożliwiają wielokrotne dołączanie tego samego pliku nagłówkowego, lub jego fragmentu podczas kompilacji wieloplikowej:

```
#ifndef nazwa  
    #define nazwa  
    ... // ← te deklaracje kompilator analizuje tylko raz,  
    ... //    podczas definiowania makro nazwa  
#endif
```

PRZYKŁAD 1. Implementacja zadania w postaci programu jednoplikowego

zawartość pliku "program.cpp"

```
// definicja podstawowego typu "Student" wspólnego dla wszystkich funkcji
struct Student
{
    char nazwisko[30];
    int wiek;
};

void Wczytaj( Student& s ); // deklaracje podstawowych operacji na strukturze "Student"
void Wswietl( const Student& s );

#define ESC      27           // deklaracje i definicje związane z interfejsem użytkownika
#define ENTER  13

#include <stdio.h>
#include <conio.h>

void WyczyscBuforConio( void );
void CzekaNaKlawisz( int klawisz );

/*=====
                               definicja głównej funkcji programu
=====*/

void main( void )
{
    Student dane_studenta;
    int klawisz;
    do
    {
        clrscr();
        printf("MENU:\n");
        printf(" 1 - wczytaj dane studenta\n");
        printf(" 2 - wyswietl dane studenta\n");
        printf("ESC - zakoncz program\n");
        klawisz=getch();
        switch( klawisz )
        {
            case '1': Wczytaj(dane_studenta); break;
            case '2': Wswietl(dane_studenta); break;
        }
    }
    while(klawisz!=ESC);
    printf("\nKoniec programu. Nacisnij klawisz ENTER");
    CzekaNaKlawisz(ENTER);
}
```

```
/*=====
      cd. pliku program.cpp - implementacje funkcji skladowych
=====*/
```

```
void Wczytaj( Student& s )
{
    clrscr();
    printf( "Operacja wczytywania danych studenta\n\n" );
    printf( "Podaj nazwisko: " );
    scanf( "%s", s.nazwisko );
    printf( "Podaj wiek: " );
    scanf( "%d", &s.wiek );
}
```

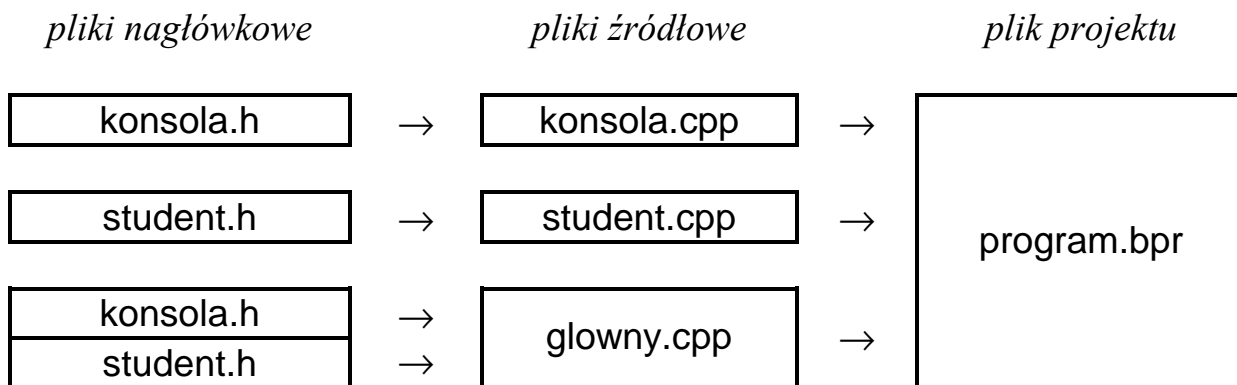
```
void Wyswietl( const Student& s )
{
    clrscr();
    printf( "Wyswietlanie danych studenta\n\n" );
    printf( "Nazwisko = %s\n", s.nazwisko );
    printf( "Wiek    = %d\n", s.wiek );
    printf( "\nNacisnij klawisz ENTER" );
    CzekaNaKlawisz( ENTER );
}
```

```
void WyczyscBuforConio( void )
{
    while( kbhit() )
        getch();
}
```

```
void CzekaNaKlawisz( int klawisz )
{
    WyczyscBuforConio();
    while( getch()!=klawisz );
}
```

```
// koniec pliku program.cpp
```

PRZYKŁAD 2. Implementacja przykładu 1 w postaci projektu wieloplikowego



zawartość pliku "konsola.h"

```
#ifndef _KONSOLA_H
#define _KONSOLA_H
// deklaracje i definicje związane z interfejsem użytkownika
#include <stdio.h>
#include <conio.h>
#define ESC 27
#define ENTER 13
void WyczyscBuforConio( void );
void CzekajNaKlawisz( int klawisz );
#endif
```

zawartość pliku "konsola.cpp"

```
#include "konsola.h"

void WyczyscBuforConio( void )
{
    while( kbhit() )
        getch();
}

void CzekajNaKlawisz( int klawisz )
{
    WyczyscBuforConio();
    while( getch() != klawisz );
}
```

zawartość pliku "student.h"

```
#ifndef _STUDENT_H
#define _STUDENT_H
#include "konsola.h"
// definicja podstawowego typu "Student" wspolnego dla wszystkich funkcji
struct Student
{
    char nazwisko[30];
    int wiek;
};
// deklaracje podstawowych operacji na strukturze "Student"
void Wczytaj(Student& s);
void Wswietl(const Student& s);
#endif
```

zawartość pliku "student.cpp"

```
#include "student.h"
#include "konsola.h"

void Wczytaj( Student& s )
{
    clrscr();
    printf( "Operacja wczytywania danych studenta\n\n" );
    printf( "Podaj nazwisko: " );
    scanf( "%s", s.nazwisko );
    printf( "Podaj wiek: " );
    scanf( "%d", &s.wiek );
}

void Wswietl( const Student& s )
{
    clrscr();
    printf( "Wswietlanie danych studenta\n\n" );
    printf( "Nazwisko = %s\n", s.nazwisko );
    printf( "Wiek    = %d\n", s.wiek );
    printf("\nNacisnij klawisz ENTER");
    CzekaNaKlawisz( ENTER );
}
```

zawartość pliku "glowny.cpp"

```
#include "student.h"
#include "konsola.h"

// ponowne dołączenie tych samych definicji spowoduje błędy
// #include "student.h" // np. "Error: multiple declaration for 'Student'"
// aby ich uniknąć należy dodać dyrektywy kompilacji warunkowej
// #ifndef, #ifdef, #else, #endif

void main(void)
{
    Student dane_studenta;
    int klawisz;
    do
    {
        clrscr();
        printf("MENU:\n");
        printf(" 1 - wczytaj dane studenta\n");
        printf(" 2 - wyswietl dane studenta\n");
        printf("ESC - zakoncz program\n");
        klawisz=getch();
        switch(klawisz)
        {
            case '1': Wczytaj(dane_studenta); break;
            case '2': Wyswietl(dane_studenta); break;
        }
    }
    while(klawisz!=ESC);

    printf("\nKoniec programu. Nacisnij klawisz ENTER");
    CzekaNaKlawisz(ENTER);
}
```

plik projektu "program.bpr"

konsola.cpp

student.cpp

glowny.cpp