Podstawy programowania komputerów

Wykład 2: „Elementarz ANSI C”

swalover@ie.tu.koszalin.pl

Historia języka C

- C powstał w Laboratoriach Bella.
- Jego poprzednikiem był interpretowany język B który Dennis Ritchie rozwinął w język C w okresie 1969-1973.
- W roku 1973 w języku C udało się zaimplementować jądro (kernel) systemu operacyjnego Unix.
- W 1978 roku Brian Kernighan i Dennis Ritchie opublikowali dokumentację języka p.t. „C Programming Language”.

- C stał się popularny po 1980 roku, i stał się dominującym językiem do programowania systemów operacyjnych i aplikacji. Na bazie języka C w latach osiemdziesiątych Bjarne Stroustrup stworzył język C++, który wprowadza możliwość programowania obiektowego.

Wikipedia

Standardy języka C

- Standard ANSI X3.159-1989 "Programming Language C". American National Standards Institute - instytucja ustalająca normy techniczne obowiązujące w USA. ANSI nie jest agencją rządową, jej działalność jest jednak silnie sprzężona z władzami USA.

- W roku 1990 standard języka C został zapisany w normie ISO 9899. Wydanie tego dokumentu było modyfikacją standardu ANSI. Język zgodny z tą wersją standardu określany jest nieformalnie jako C89.

- Od tego czasu powstało wiele uaktualnień tej normy. Ostatnia wersja ma oznaczenie ISO 9899:1999, język zgodny z tą normą określany jest nieformalnie jako C99. C99 nie jest kompatybilny z C++.

Wikipedia

Elementy języka C

- użytkowane symbole
- identyfikatory
- zastrzeżone słowa kluczowe

Użytkowane symbole

Tabela kodów ASCII, zgodnych z międzynarodowym kodem ISO-7, zawiera:

- symbole alfanumeryczne (cyfry, litery)
- znaki przestankowe
- inne znaki występujące w tekstach (apostrof,gwiazdka itp.)
- znaki niedrukowalne sterujące pracą urządzeń

swalover@ie.tu.koszalin.pl

Identyfikatory

- to słowa, którymi oznaczamy obiekty programu źródłowego
- identyfikator składa się z liter, cyfr oraz znaku podkreślenia

  _ (underline)
- litery małe i wielkie uważa się za różne

  INDEX, index, InDeX
- za znaczące uważa się pierwsze 32 znaki

Zastrzeżone słowa kluczowe

Standard ANSI C:

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

Dodatkowo

near	far	huge

Elementy programu C

- dyrektywy preprocesora
- komentarze
- deklaracje
- instrukcje

Dyrektywy preprocesora

```
#include<nazwa_pliku>
```

header file

```
#include <stdio.h>
```

standard input/output header file

```
#include<iostream.h>
```

strumienie cin i cout

Komentarz

Tekst objaśniający, ignorowany przez komputer pod czas wykonywania poleceń

- `/* Komentarz */` - dowolny tekst jednowierszowy lub wielowierszowy
- `// Komentarz` - tekst jednowierszowy, metoda nie jest częścią standardu ANSI C

Deklaracja

Polecenie przypisujące stałej wartość lub łączące zmienną z określonym adresem w pamięci oraz typem danych

```
int i, j, k;
char znak;
float x;
short nr, Nr, licznik;
```

Instrukcje

Ich można skompilować do postaci ciągu poleceń języka maszynowego.

Typy **instrukcji**: przypisania, sterujące, skoku, warunkowe, wyboru i pętli.

- w kodzie źródłowym C instrukcja zawsze zakończona jest średnikiem
- instrukcja złożona to ciąg instrukcji zgrupowanych przy użyciu nawiasów { ... }
- dla zapisu instrukcji stosowane są operatory

Operatory

To nazwa lub symbol używane do określenia rodzaju operacji wykonywanej na danych (operandach)

Wyróżniamy operatory:

- przypisania: `=` `+=` `-=` `*=` `/=` `%=`
- arytmetyczne: `+` `-` `*` `/` `%` `++` `--`
- relacji: `>` `>=` `<` `<=` `==` `!=`
- logiczne: `&&` `||` `!`
- wprowadzania i wyprowadzania: `>>` `<<` (z C++)

Wyrażenia

Są arytmetyczne i logiczne.

Kolejność wykonania operacji arytmetycznych:

1. ( )
2. ++ --
3. * / %
4. + -
5. = += -= *= /= %=

Struktura programu C

Program zbudowany jest z funkcji.

Program musi zawierać funkcję **main ( )**

```
int main(void)
     {
     /*tu się mieści merytoryczna
część programu*/
     return 1;
     }
```

Przed funkcją `main()` mogą być umieszczone komentarze oraz dyrektywy preprocesora

Funkcja

W języku programowania to nazwana i zachowana procedura zwracająca jedną wartość.

Definicja funkcji ma postać:

```
<typ_wyniku><nazwa_funkcji>(<arg>)
{
<deklaracje i instrukcje>
}
```

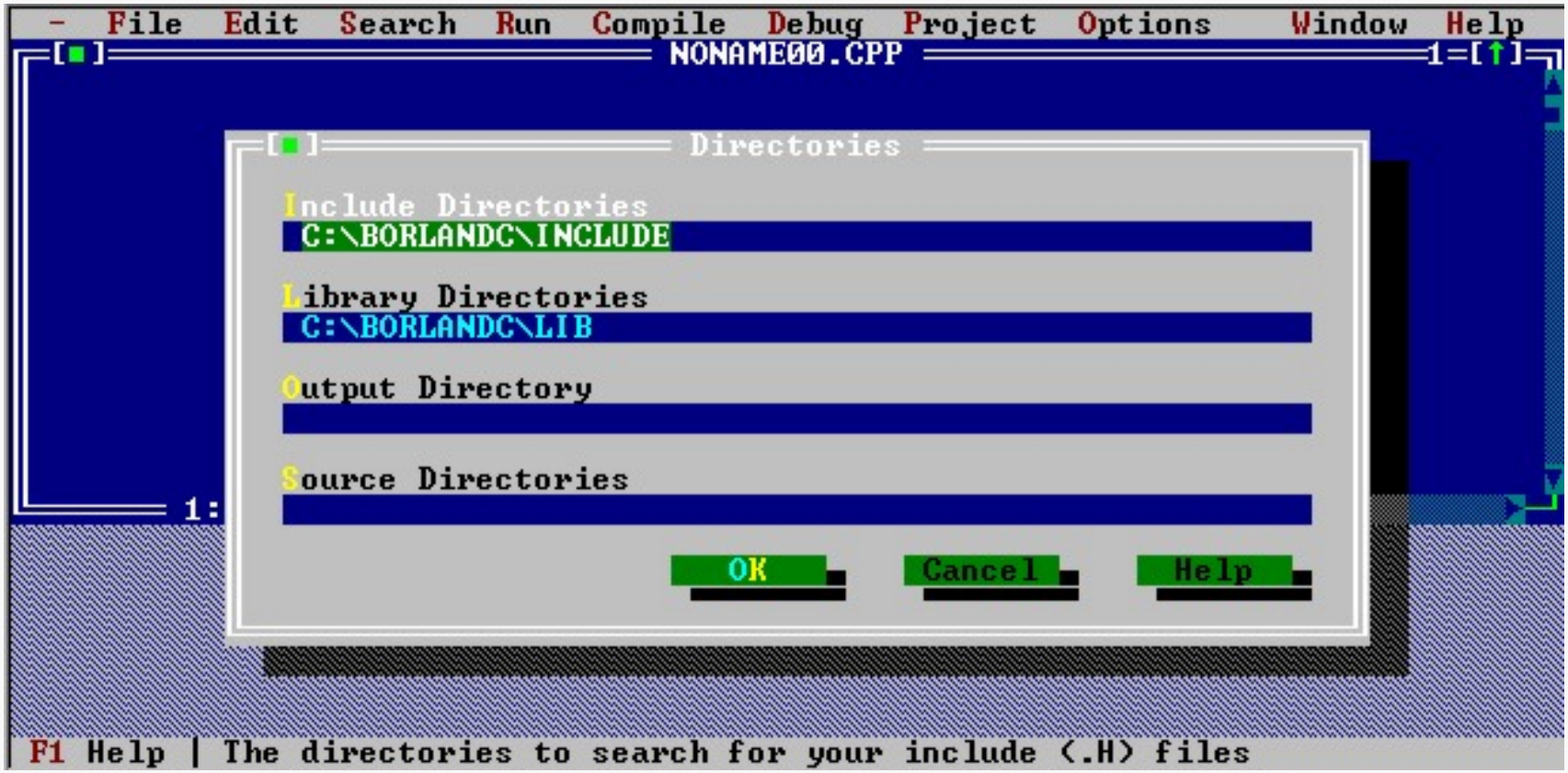
Funkcje biblioteczne

Najczęściej używane funkcje zostały zdefiniowane w bibliotekach języka C.

Przykład – funkcje biblioteki **stdio.h**

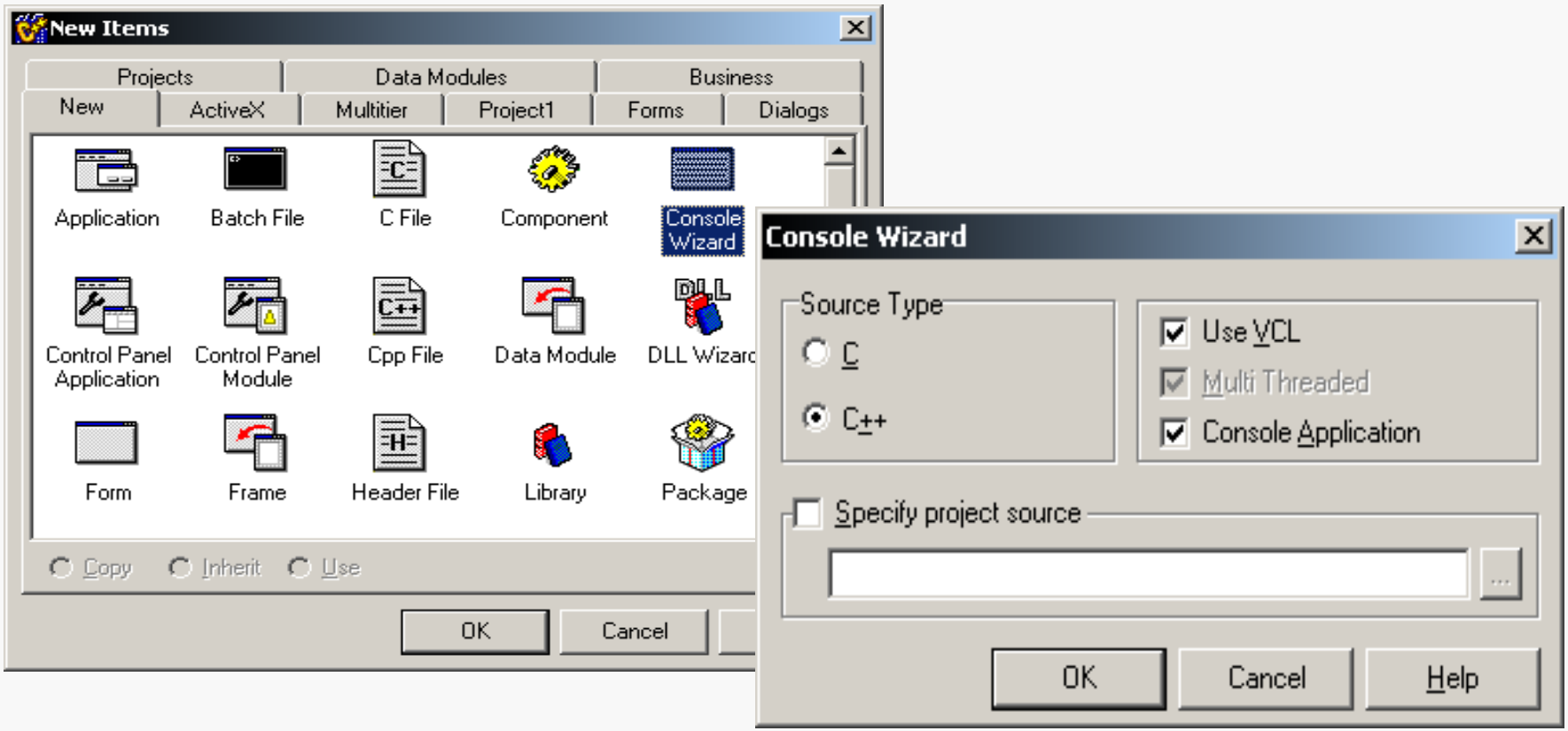
- `getchar()` - pobierz jeden znak z stdin
- `putchar ()` - wyprowadź znak do stdout
- `gets(), puts()` - wpr./wypr. łańcuch
- `scanf(), printf()` - wpr./wypr. z formatowaniem

Zmienne środowiskowe w Borland C++



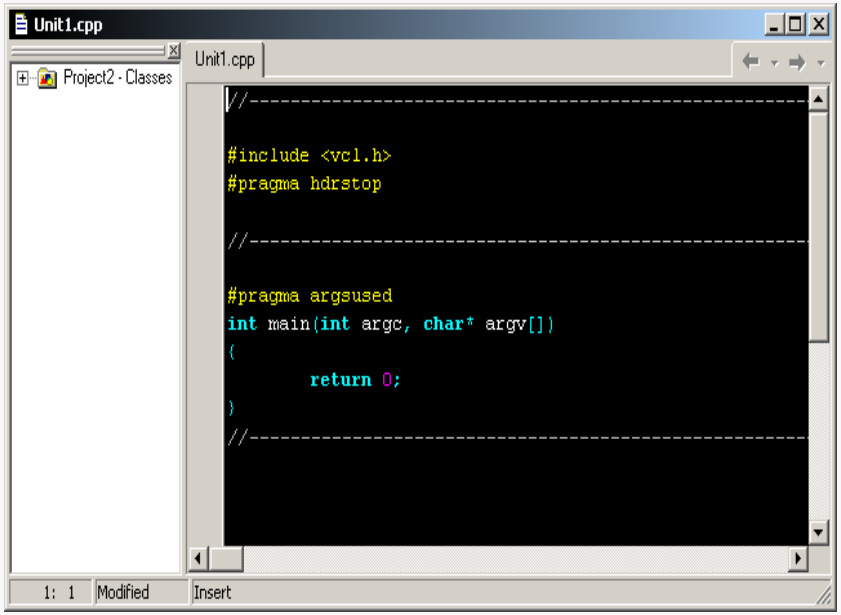
Menu **Options** polecenie **Directories** - ścieżki dostępu dla środowiska Borland.

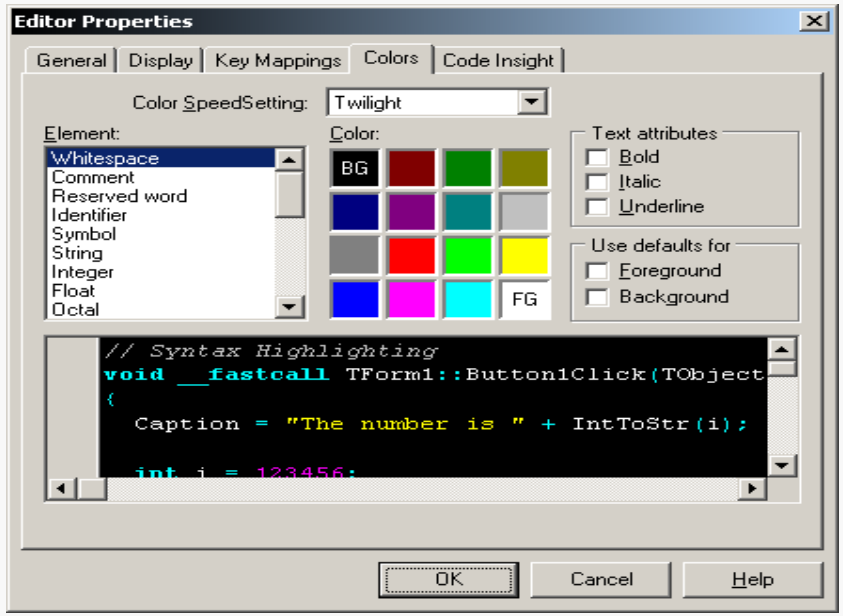
C++ Builder dla Windows



Praca w 32bitowej konsoli: **File** – **Close All**, **New** – **New Items** – **Console Wizard**.

Praca z Builderem





Kompilować - F9, uruchomić - Run.