

Header conio.h

Header	Opis zawartych funkcji	Przykładowe funkcje
conio.h	Operacje wejścia i wyjścia dla konsoli	system("cls"), kbhit(), getch(), getche(), putch(), ungetch()

#include<conio.h>

conio <-- console input & output

- **system("cls")** - nie z conio.h Dev C++ ale pasuje do tematu bo wyczyszcza ekran pozostawiając go pustym, podobnie jak to robi funkcja **clrscr()** z conio.h Turbo C++
- **kbhit()** - rozpoznaje czy jakikolwiek klawisz klawiatury został ostatnio przyciśnięty. Jeżeli wartość tej funkcji jest w tym miejscu niezerowa, to na wartość przyciśnięcia klawisza czeka bufor klawiatury. Wtedy wywołanie funkcji **getch()** lub **getche()** powoduje otrzymanie tej wartości
- **getch()** - przytrzymuje ekran aż do wprowadzenia dowolnego znaku. Wprowadzonego znaku nie widzimy na ekranie
- **getche()** - to samo co **getch()** ale wypisuje znak na ekranie
- **putch(znak)** - umieszcza znak na ekranie
- **ungetch(znak)** - powoduje cofnięcie ostatnio odczytanego z klawiatury przez **getche()** znaku z powrotem do konsoli (bufora klawiatury) w wyniku czego następnym odczytanym znakiem jest właśnie ten znak

```
// Operacje wejścia i wyjścia dla konsoli (klawiatury i ekranu)

#include <stdio.h>
#include <conio.h>           // ten temat
#include <locale.h>          // dla 'setlocale()'

char bufor[50];
int indeks = 0;
char znak;

int main()
{ setlocale(LC_CTYPE, "Polish"); //polskie znaki
  printf("\n Gdy program się zatrzyma, przyciśnij dowolny klawisz.\n");

  printf("\n ----- kbhit() ----- \n");

  if ( kbhit() )
  { printf("Przycisnij dowolny klawisz aby wyjść z instrukcji if\n");
    getch();
  }
  else
    printf(" Wartość zwrócona przez kbhit() to 0 \n");

  printf("\n ----- getch() i getche() ----- \n");

  printf("\n Test na getch(): "); znak = getch();
```

```

// printf("\n Przyjąłem znak %c", znak);
printf("\n Przyjąłem znak %c");
putch(znak);      // odpowiednik   putc(znak, stdout);

printf("\n\n Test na getche(): "); znak = getche(); // getch() jest w 'conio.h'
// printf("\n Przyjąłem znak %c", znak);
printf("\n Przyjąłem znak %c");
putch(znak);      // odpowiednik   putc(znak, stdout);

printf("\n\n ----- ungetch() ----- \n");

printf(" Wpisz wyrażenie zakończone przecinkiem : ");
znak = getche();
while( indeks < 49 )      // musi być miejsce na indeks nr 50 zamykający łańcuch
{ if( znak == ',' )      // przecinek kończy zapis zmiennej (jak w plikach .csv)
    break;
    bufor[indeks++] = znak;
    znak = getche();
}
ungetch( znak );          // wycofaj ostatnio przeczytany znak do konsoli
bufor[indeks] = '\0';     // zamknij wpisany łańcuch znakowy
printf("\n Przeczytane: %s\n", bufor );
printf(" Następne czytanie: %c", getch() );
printf("\ngetch() przeczytał ponownie przecinek bo on został przedtem przez \
ungetch() z powrotem wrzucony do konsoli (tu: bufora klawiatury).");

return 0;
}

```

Wynik działania programu:

```

Gdy program się zatrzyma, przyciśnij dowolny klawisz.

----- kbhit() -----
Wartość zwrócona przez kbhit() to 0

----- getch() i getche() -----

Test na getch():
Przyjąłem znak  k

Test na getche(): W
Przyjąłem znak  W

----- ungetch() -----
Wpisz wyrażenie zakończone przecinkiem : Ala ma kota,
Przeczytane: Ala ma kota
Następne czytanie: ,
getch() przeczytał ponownie przecinek bo on został przedtem przez ungetch() z powrotem
wrzucony do konsoli (tu: bufora klawiatury).
-----
Process exited after 154.4 seconds with return value 0
Press any key to continue . . . _

```