05

Języki programowania 1

Zapis i odczyt informacji – obsługa standardowego wejścia/wyjścia.

Zakres:

- getc() (wczytaj znak),
- putc() (drukuj znak),
- getchar() (pobierz znak),
- putchar() (wyślij znak).

Co to jest Standard Input/Output?

Wszystkie pliki zawierające jakiś program przez komputer identyfikowane są jako ciąg binarny. Język C plik traktuje, jako ciąg bajtów, który może być nazywany plikiem lub też strumieniem. C traktuje wszystkie strumienie w zasadzie tak samo, choć mogą one przepływać z różnych kierunków. Dodatkowo w C są trzy strumieniowe pliki (file stream), które standardowo są otwierane wstępnie:

- stdin – standardowe wejście do odczytu (pre-opened standard input),
- stdout – standardowe wyjście do zapisu (pre-opened standard output),
- stderr – standardowe wyjście diagnostyczne, dla komunikatów o błędach (standard error).

Zazwyczaj stdin zostaje powiązany z klawiaturą, a stdout i stderr zostają powiązane z ekranem (monitora, terminala). Używana funkcja printf(), przesyła dane do standardowego strumienia wyjściowego stdout, który kierowany jest na ekran.

Język C zawiera wiele funkcji do odczytu i zapisu we/wy. Plik nagłówkowy stdio.h zawiera deklaracje tych funkcji. Dlatego zawsze operując na we/wy należy dołączyć do programu plik nagłówkowy stdio.h.

Wczytywanie danych wejściowych od użytkownika

Standardowymi operacjami do wprowadzania danych jest wpisywanie znaków z klawiatury. Zazwyczaj biblioteki dostarczone wraz z kompilatorami oferują wiele funkcji operacji wejścia/wyjścia. Przykładowymi mogą być funkcje:

- **getc()** funkcja ta wczytuje pojedynczy znak ze strumienia (pliku) wejściowego i zwraca numer, czy też kod wczytanego znaku w formacie liczby całkowitej (**Program getc.c**),
- **getchar()** ma bardzo podobne zadanie jak funkcja getc(), dokładniej getc(stdio) realizuje to samo co funkcja getchar(); funkcji tej nie trzeba przekazywać żadnych argumentów, zwraca ona wartość numeryczną całkowitą ze strumienia stdin (**modyfikacja programu getc.c**)

Drukowanie na standardowym urządzeniu wyjścia (ekranie)

Dualnie do funkcji getc() getchar() język C zawiera funkcje putc() i putchar() do wprowadzania znaku ze standardowego strumienia wyjściowego.

- Funkcja **putc()** wysyła znak do wskazanego jej jako argument strumienia/pliku (np. do standardowego strumienia wyjściowego stdout, czyli na ekran monitora) **program putc.c**,
- Funkcja **putchar()** podobnie jak poprzednia używana jest do wyprowadzania znaku na ekran. Jedyna różnica między nimi polega na tym, że putchar() potrzebuje tylko jednego argumentu – znaku przeznaczonego do wyprowadzenia. Nie trzeba podawać strumienia,

ponieważ funkcja ta przyjmuje domyślnie strumień standardowego wyjścia stdout. (**modyfikacja programu putc.c**).

Funkcja printf()

Jedną z ważniejszych i bardzo często używanych funkcji w języku C jest funkcja printf(). Postać jej w uproszczeniu można podać jako:

printf(”Liczba zmiennoprzecinkowa: %f; Liczba całkowita: %d”, 15.3,258);

łańcuch formatujący wyrażenia.

Należy pamiętać, że przy wywołaniu funkcji printf() należy użyć dokładnie tej samej ilości wyrażeń, ile zostało wcześniej użytych specyfikatorów formatu.

Funkcja printf() zwraca ilość poprawnie (zakończonych sukcesem) sformatowanych wyrażeń. Jeżeli nastąpi błąd, to funkcja zwraca wartość ujemną.

Lista specyfikatorów:

Specyfikator	**Opis**
%c	Znak – typu char
%d	Liczba – typu int
%i	Liczba – typu int
%f	Liczba zmiennoprzecinkowa – typ float
%e	Format wykładniczy z użyciem małej litery
%E	Format wykładniczy z użyciem dużej litery
%g	Zastosuj %f lub %e – wybierz format, którego wynik będzie krótszy
%G	Zastosuj %f lub %E – wybierz format, którego wynik będzie krótszy
%o	Liczba ósemkowa bez znaku
%s	Łańcuch znaków (string)

%u	Liczba całkowita bez znaku (unsigned)
%x	Liczba szesnastkowa bez znaku (z zastosowaniem przyrostka x, małe litery)
%X	Liczba szesnastkowa bez znaku (z zastosowaniem przyrostka x, duże litery)
%p	Argument odpowiadający wskaźnikowi (pointer)
%n	Rejestruj ilość znaków wprowadzona do tego momentu
%%	Wprowadź znak % (jawnie)

Zadanie: zamiana liczb dziesiętnych na szesnastkowe.

Specyfikacja minimalnej szerokości pola wydruku, wyrównanie pola wyjściowego oraz specyfikator precyzji

Język C pozwala na wstawianie w specyfikatorach formatu pomiędzy znak % a literę, liczby całkowitej. Liczba taka nazywana jest *specyfikatorem minimalnej szerokości pola*. Liczba ta określa minimalną szerokość tj. ilość znaków w polu wyjściowym.

Przykład określania minimalnego pola wyjściowego: *pole*.c

Zazwyczaj znaki wyjściowe są w obrębie pola wyrównywane do prawej. Można to jednak zmienić poprzedzając dany specyfikator znakiem minus (-). Wówczas będzie wyrównany do lewej.