Politechnika Białostocka
Wydział Elektryczny
Katedra Elektrotechniki, Energoelektroniki i Elektroenergetyki

Instrukcja
do pracowni specjalistycznej z przedmiotu

Programowanie C

Kod przedmiotu: **CP1S01005**

(studia stacjonarne)

JĘZYK C - STRUKTURY

Numer ćwiczenia

PRC_12

Autor:
dr inż. Jarosław Forenc

Białystok 2024

Spis treści

Materiały dydaktyczne przeznaczone dla studentów Wydziału Elektrycznego PB.

1. Opis stanowiska

1.1. Stosowana aparatura

Podczas zajęć wykorzystywany jest komputer klasy PC z systemem operacyjnym Microsoft Windows 10/11 oraz platforma Arduino wraz z zestawem czujników.

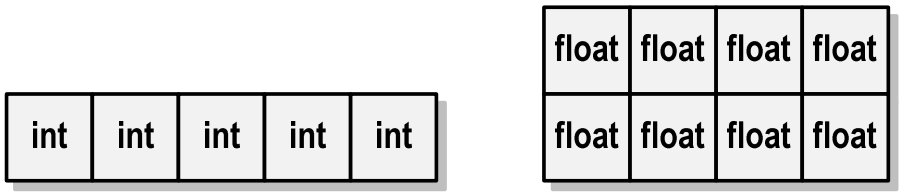
1.2. Oprogramowanie

Na komputerach zainstalowany jest edytor kodu źródłowego Visual Studio Code 1.92 (lub nowszy) wraz z rozszerzeniem (PlatformIO IDE for VSCode).

2. Wiadomości teoretyczne

2.1. Struktury

W języku C do przechowywania wielu elementów tego samego typu stosuje się tablice jedno- lub wielowymiarowe (Rys. 1).



Rys. 1. Reprezentacja tablicy jednowymiarowej (wektora)
i tablicy dwuwymiarowej (macierzy) w języku C

W przypadku, gdy pod jedną nazwą mają być zgrupowane elementy różnych typów, stosuje się **struktury**. Struktury służą zatem do reprezentacji złożonych obiektów różnych danych (Rys. 2).

Rys. 2. Reprezentacja struktury w języku C

Ogólna postać deklaracji struktury jest następująca:

```
struct nazwa_struktury
{
    typ nazwa_pola_1;
    typ nazwa_pola_2;
    ...
    typ nazwa_pola_n;
};
```

Deklaracja struktury rozpoczyna się od słowa kluczowego **struct**, po którym może występować opcjonalna **nazwa struktury** (**etykieta**). Pomiędzy nawiasami klamrowymi umieszczone są **pola struktury** (**dane**, **komponenty**, **składowe**), które mają taką samą postać jak deklaracje zmiennych w programie. Deklaracja struktury kończy się średnikiem umieszczonym po nawiasie klamrowym zamykającym. Brak tego średnika jest najczęściej popełnianym błędem przy deklaracji struktury.

Poniżej przedstawione są przykłady deklaracji dwóch struktur. Struktura **punkt** przechowuje współrzędne **x** i **y** punktu w prostokątnym układzie współrzędnych. Struktura **data** opisuje datę w kalendarzu, przechowując **dzień, miesiąc** i **rok** jako liczby całkowite typu **int**.

```
struct punkt
{
    float x;
    float y;
};
```

```
struct data
{
    int dzien;
    int miesiac;
    int rok;
};
```

Przy nazywaniu pól struktury obowiązują takie same zasady, jak przy nadawaniu nazw innym identyfikatorom. Nazwy pól struktury mogą być takie same jak nazwy innych zmiennych w programie, a nawet takie same jak nazwa struktury. W różnych strukturach mogą występować pola o identycznych nazwach. Pola tego samego typu mogą być umieszczone w jednym wierszu (np. **int dzien, miesiac, rok;**).

Deklarując strukturę, wprowadzamy **nowy typ danych** (np. **struct data**), którym możemy posługiwać się tak samo jak każdym innym typem standardowym. Deklaracja struktury nie tworzy obiektu (nie przydziela pamięci na pola), a jedynie określa, z czego się on składa. Aby do struktury można było zapisać dane, należy zdefiniować **zmienną strukturalną**.

Definicja zmiennej strukturalnej może być połączona z deklaracją struktury. W takim przypadku nazwy zmiennych strukturalnych umieszcza się bezpośrednio po klamrze zamykającej listę pól struktury (np. zmienna **pom1** w poniższym przykładzie). Deklaracje zmiennych strukturalnych mogą być również umieszczone później w kodzie programu. Takie deklaracje mają identyczną postać jak deklaracje zmiennych standardowych typów (np. **struct pomiar** - nazwa typu, **pom2** - nazwa zmiennej).

```
#include <Arduino.h>

struct pomiar {
  float cisnienie;
  float temperatura;
} pom1;


void setup() {
  struct pomiar pom2;
}

void loop() {
}
```

Deklaracja struktury może znajdować się w dowolnym miejscu programu. Należy jednak pamiętać, że umieszczenie jej wewnątrz definicji funkcji ograniczy jej widoczność wyłącznie do tej funkcji. W powyższym programie deklaracja struktury **pomiar** znajduje się na początku programu. Jest to sytuacja najczęściej spotykana w praktyce.

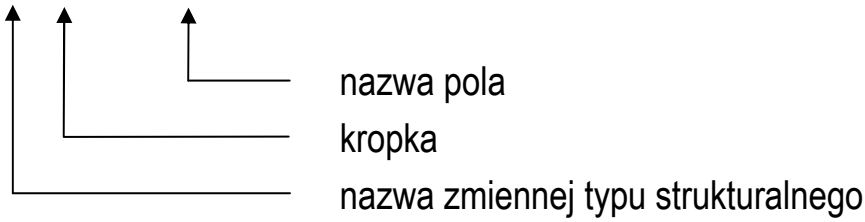
2.2. Odwołania do pól struktury

Dostęp do pól struktury jest możliwy dzięki konstrukcji typu:

```
nazwa_zmiennej_strukturalnej.nazwa_pola
```

Operator kropki nazywany jest operatorem bezpośredniego wyboru pola. W poniższym przykładzie zapisujemy wartość **25** do pola **temperatura** zmiennej strukturalnej **pom1**.

```
pom1.temperatura = 25;
```



Wyrażenie **pom1.temperatura** traktowane jest jak zmienna typu **int**.

W przypadku posługiwania się **wskaźnikiem** do zmiennej strukturalnej, odwołania do pól struktury wymagają użycia operatora pośredniego wyboru pola, zapisywanego w postaci znaku minus i znaku większości (**->**), np.

```
struct pomiar pom1, *Ptr_pom1;
Ptr_pom1 = &pom1;
Ptr_pom1 -> temperatura = 25;
```

lub

```
(*Ptr_pom1).temperatura = 25;
```

W ostatnim zapisie dodatkowe nawiasy są konieczne, ponieważ operator kropki (**.**) ma wyższy priorytet niż operator de referencji (*).

2.3. Inicjalizacja zmiennej strukturalnej

W deklaracjach pól struktury nie mogą występować inicjalizacje. Można natomiast inicjalizować zmienne strukturalne, np.

```
struct pomiar pom1 = {1034,25};
```

Kolejność wartości w nawiasach klamrowych musi być zgodna z kolejnością deklaracji pól struktury. Brakujące pola w inicjalizacji są zastępowane zerami. Każda wartość początkowa musi być wyrażeniem stałym (nie może być zmienną).

Zmienne strukturalne tego samego typu można sobie przypisywać (nawet jeśli zawierają tablice znaków), ale nie można ich porównywać ze sobą.

```
struct pomiar pom1 = {1034,25};
struct pomiar pom2;
pom2 = pom1;
```

3. Przebieg ćwiczenia

Na pracowni specjalistycznej należy wykonać wybrane zadania wskazane przez prowadzącego zajęcia. W różnych grupach mogą być wykonywane różne zadania.

1. Zdefiniuj strukturę opisującą wybrane **urządzenie elektryczne**. Struktura powinna składać się z co najmniej trzech pól różnych typów. Następnie zadeklaruj dwie zmienne strukturalne. Pierwszą z nich zainicjalizuj wartościami, a drugiej przypisz wartości. Wyświetl w oknie monitora portu szeregowego zawartość obu zmiennych.

2. Zdefiniuj strukturę składającą się z dwóch pól opisujących częstotliwość i czas trwania dźwięku. Następnie zadeklaruj tablicę takich struktur i zainicjalizuj ją dźwiękami tworzącymi wybraną melodię. Po uruchomieniu programu,

naciśnięcie przycisku SW1 powinno spowodować odtworzenie melodii na buzzerze.

3. Zdefiniuj strukturę zawierającą wyniki pomiaru ciśnienia atmosferycznego i temperatury powietrza. Zadeklaruj 10-elementową tablicę takich struktur. Po uruchomieniu programu do tablicy powinno zostać zapisanych 10 pomiarów wykonanych co 5 sekund. Oblicz i wyświetl na wyświetlaczu OLED średnie ciśnienie i średnią temperaturę.

4. Literatura

[1] Prata S.: Język C. Szkoła programowania. Wydanie VI. Helion, Gliwice, 2016.

[2] Wrotek W.: Arduino od podstaw. Helion, Gliwice, 2023.

[3] Monk S.: Arduino dla początkujących. Podstawy i szkice. Helion, Gliwice, 2019.

[4] Evans M., Noble J., Hochenbaum J.: Arduino w akcji. Helion, Gliwice, 2014.

[5] Kernighan B.W., Ritchie D.M.: Język ANSI C. Programowanie. Wydanie II. Helion, Gliwice, 2010.

[6] https://code.visualstudio.com/ - Visual Studio Code

[7] https://www.arduino.cc/reference/en/ - Arduino Language Reference

5. Pytania kontrolne

1. Omów sposób deklarowania struktur w języku C.
2. W jaki sposób można odwoływać się do pól struktury?
3. Opisz inicjalizację zmiennych strukturalnych.

6. Wymagania BHP

Warunkiem przystąpienia do praktycznej realizacji ćwiczenia jest zapoznanie się z instrukcją BHP i instrukcją przeciw pożarową oraz przestrzeganie zasad w nich zawartych.

W trakcie zajęć laboratoryjnych należy przestrzegać następujących zasad.

- Sprawdzić, czy urządzenia dostępne na stanowisku laboratoryjnym są w stanie kompletnym, nie wskazującym na fizyczne uszkodzenie.
- Jeżeli istnieje taka możliwość, należy dostosować warunki stanowiska do własnych potrzeb, ze względu na ergonomię. Monitor komputera ustawić w sposób zapewniający stałą i wygodną obserwację dla wszystkich członków zespołu.
- Sprawdzić prawidłowość połączeń urządzeń.
- Załączenie komputera może nastąpić po wyrażeniu zgody przez prowadzącego.
- W trakcie pracy z komputerem zabronione jest spożywanie posiłków i picie napojów.
- W przypadku zakończenia pracy należy zakończyć sesję przez wydanie polecenia wylogowania. Zamknięcie systemu operacyjnego może się odbywać tylko na wyraźne polecenie prowadzącego.
- Zabronione jest dokonywanie jakichkolwiek przełączeń oraz wymiana elementów składowych stanowiska.
- Zabroniona jest zmiana konfiguracji komputera, w tym systemu operacyjnego i programów użytkowych, która nie wynika z programu zajęć i nie jest wykonywana w porozumieniu z prowadzącym zajęcia.
- W przypadku zaniku napięcia zasilającego należy niezwłocznie wyłączyć wszystkie urządzenia.
- Stwierdzone wszelkie braki w wyposażeniu stanowiska oraz nieprawidłowości w funkcjonowaniu sprzętu należy przekazywać prowadzącemu zajęcia.

- Zabrania się samodzielnego włączania, manipulowania i korzystania z urządzeń nie należących do danego ćwiczenia.
- W przypadku wystąpienia porażenia prądem elektrycznym należy niezwłocznie wyłączyć zasilanie stanowiska. Przed odłączeniem napięcia nie dotykać porażonego.