

Ćwiczenia: język C lista zadań nr 2: kompilator gcc, style formatowania kodu

1. Zapoznaj się z kompilatorem gcc (<https://gcc.gnu.org/>, wersja ≥ 8.0).
Zapisz w pliku *hello.c* poniższy kod programu:

```
1 void main()  
2 {  
3     printf("hello_world\n");  
4 }
```

- (a) Skompiluj powyższy kod (bez dodatkowych opcji), a następnie z opcjami `-Wpedantic -Wall -Wextra`
Jakie ostrzeżenia pojawiają się podczas kompilacji? Dlaczego?
Popraw podany kod tak, ażeby żadne ostrzeżenia się nie pojawiały
 - (b) Skompiluj ponownie powyższy kod dodając do opcji kompilacji z pkt (a) zachowanie odpowiedniego standardu języka, tzn. z opcją:
`-std=c90`
a następnie z opcją `-std=c99`, `-std=c11` i ostatecznie z `-std=c18` Za każdym razem popraw program tak, aby nie pojawiało się żadne ostrzeżenie. Czy jest jakaś zauważalna różnica? Jaka jest wartość kodu zakończenia programu? (sprawdź poprzez wydanie polecenia `echo $?`). Dlaczego właśnie taka?
 - (c) Dodaj komentarz opisujący każdą linię kodu.
2. Zapoznaj się z dokumentacją programu *indent* (<https://www.gnu.org/software/indent/>).
Sformatuj kod programu *hello.c* używając domyślnego stylu formatowania, oraz wypróbuj:
 - (a) standard K&R: `-kr`
 - (b) berkeley: `-orig`
 - (c) linux kernel (należy użyć wszystkich przełączników na raz): `-nbad -bap -nbc -bbo -hnl -br -brs -c33 -cd33 -ncdb -ce -ci4 -cli0 -d0 -di1 -nfc1 -i8 -ip0 -l80 -lp -npcs -nprs -npsl -sai -saf -saw -ncs -nsc -sob -nfca -cp33 -ss -ts8 -il1`
 3. Jakie wyniki da poniższy program? Popraw zauważone błędy.

```
1 #include <stdio.h>  
2  
3 int main (void)  
4 {  
5     printf ("Testowanie...");  
6     printf ("...1");  
7     printf (...2)  
8     printf (...3);  
9     printf (\n");  
10    ....return 0;  
11 }
```

4. Napisz program, który wyświetli na ekranie terminala następujące napisy:
 - (a) W języku C wielkość liter ma znaczenie.
 - (b) Wykonanie programu zaczyna się od funkcji main.
5. Napisz program odejmujący od 87 wartość 15 i wyświetlający wynik. Kod źródłowy powinien zawierać stosowny opis.
6. W poniższym programie znajdź błędy składniowe, następnie przepisz i uruchom poprawiony program, żeby upewnić się, że wszystkie błędy zostały zlokalizowane.

```
1 #include <stdio.h>
2
3 int main (Void)
4 {
5     INT sum
6     /* wylicz wynik
7     sum = 25 + 37 - 119
8     /* pokaz wyniki //
9     printf ("Odpowiedz: %i\n" sum);
10    return 0;
11 }
```

7. Jaki wynik da poniższy program? Dodaj stosowny opis do kodu źródłowego.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main (void)
5 {
6     int odpowiedz, wynik;
7     odpowiedz = 100;
8     wynik = odpowiedz - 10;
9     printf ("Wynik: _%i\n", wynik - 5);
10    exit(EXIT_SUCCESS);
11 }
```

8. Napisz program *kwadrat*, wyświetlający kwadrat liczby podanej z klawiatury. Do wprowadzenia liczby użyj funkcji `scanf()`, której deklaracja znajduje się w `<stdio.h>`.

Działające programy z zadań 3–8 należy przesłać prowadzącemu.