

Funkcje

1. Zdefiniuj funkcję *przeciwprostokatna* obliczającą długość przeciwprostokątnej trójkąta prostokątnego z podanych długości przyprostokątnych. Wykorzystaj tę funkcję w programie do określania długości przeciwprostokątnej każdego z następujących trójkątów:

trójkąt	bok 1	bok2
1	3.0	4.0
2	5.0	5.0
3	1.0	6.0

Funkcja powinna pobierać dwa argumenty typu *double* i zwracać długość przeciwprostokątnej też jako *double*.

2. Napisz funkcję, która oblicza wartość silni z liczby naturalnej. Wykorzystaj ją w programie wypisującym na wyjście silnię 10 pierwszych liczb naturalnych.

3. Liczba doskonała to liczba, która jest sumą dzielników od niej mniejszych. Na przykład 6 jest liczbą doskonałą ponieważ:

$$1 + 2 + 3 = 6$$

Napisz funkcję, która sprawdza czy podana liczba jest doskonała. Użyj tej funkcji do wypisania wszystkich liczb doskonałych mniejszych od 1000.

4. Napisz funkcję, która po wczytaniu liczby całkowitej wypisze jej cyfry zaczynając od ostatniej i kończąc na pierwszej. Na przykład po wczytaniu liczby '1410' funkcja powinna wypisać '0141'.

5. Napisz program, który wyświetla prostokąt, owal, strzałkę i diament w/g poniższego wzoru:

```
*****      **      *      *
*      *      *      *      ***      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*      *      *      *      *      *      *
*****      **      *      *
```

Do każdej figury użyj osobnej funkcji, np. funkcja *prostokat* powinna dać się użyć do rysowania prostokątów różnej wielkości:

```
void prostokat(int szerokosc, int wysokosc);
```

6. Napisz program wypisujący przeskalowane liczby LCD:

```
-      -      -
|      |      |
|      -      -      itd.
|      |      |
      -      -
```

Skorzystaj z poniższego szablonu:

```
#include <stdio.h>
void skaluj(int liczba, int skala) {
    /* wypisz liczbę w podanej skali */
}
int main (int argc, char *argv[]) {
    int l, w;
    if (argc == 2) {
        /* wypisywanie podanej liczby, skala = 1 */
        l = atoi(argv[1]);
        w = 1;
    } else if (argc == 3) {
        /* wypisywanie liczby ze podanym skalowaniem */
        l = atoi(argv[1]);
        w = atoi(argv[2]);
    }
    skaluj(w, l);
    return 0;
}
```

7. Komputery odgrywają coraz większą rolę w nauczaniu. Napisz funkcję, która sprawdza umiejętność mnożenia liczb naturalnych w zakresie do 100.

Użyj tej funkcji w programie zadającym 5 pytań sprawdzających umiejętność mnożenia takich liczb. Program ma komentować odpowiedzi. Po udzieleniu odpowiedzi na ostatnie pytanie program ma wypisać liczbę odpowiedzi dobrych i złych.

8. Popraw poprzedni program tak, by w zależności od odpowiedzi na poszczególne pytania wyświetlany był losowo jeden z komunikatów: *Bardzo dobrze!*, *Świetnie*, *Dobra robota!*, *Zła odpowiedź*, *Oj, niedobrze*, *Następnym razem może pójdzie Ci lepiej*. W programie użyj instrukcji *switch*.

9. Napisz program, do gry *zgadnij liczbę*.

Oto zasady gry. Komputer losuje liczbę z zakresu od 1 do 1000, a użytkownik próbuje ją odgadnąć, otrzymując podpowiedzi: *za dużo*, *za mało*. Po zgadnięciu liczby, komputer powinien oczywiście wypisać **gratulacje, zgadłeś!**

Program ten powinien składać się z kilku funkcji, tak jak program grający w [craps](#). **Uwaga:** program *craps* korzysta z biblioteki CSLib. Dlatego kompilujemy go, tak jak to zostało podane na wykładzie, za pomocą polecenia *gccx*.