

Funkcje. Rekurencja

1. Napisz i przetestuj funkcję obliczającą średnią arytmetyczną dwóch liczb. Prototyp funkcji:

```
double avg(double a, double b);
```

2. Napisz funkcję obliczającą mniejszą z dwóch danych liczb. Prototyp funkcji:

```
int min(int a, int b);
```

3. Napisz funkcję obliczającą n -ty wyraz ciągu Fibonacciego. Prototyp funkcji:

```
int fibonacci(int n);
```

4. Napisz funkcję odwracającą napis. Prototyp funkcji:

```
string reverse(string s);
```

5. Napisz funkcję robiącą to samo, ale w miejscu. Prototyp funkcji:

```
void reverse(string &s);
```

6. Napisz funkcję zamieniającą wartości zmiennych typu char miejscami. Prototyp funkcji:

```
void swap(char &a, char &b);
```

Następnie przepisz funkcję z ćwiczenia 5 tak, by korzystała z funkcji swap.

7. Napisz funkcję sprawdzającą, czy dana liczba jest kwadratem liczby całkowitej. Prototyp funkcji:

```
bool is_square(int n);
```

8. Napisz funkcję sprawdzającą, czy dana liczba jest liczbą pierwszą. Prototyp funkcji:

```
bool is_prime(int n);
```

9. Napisz rekurencyjną funkcję `writeLine()` wypisujący w kółko zadany znak, tworząc linię złożoną z n znaków. Na przykład wywołanie

```
writeLine('*', 5)
```

wyświetla

```
*****
```

10. Napisz rekurencyjną funkcję `writeBlock()` wykorzystującą funkcję `writeLine()` z poprzedniego ćwiczenia, wypisującą m linii po n znaków w każdej. Na przykład wywołanie

```
writeBlock('*', 5, 3)
```

wyświetla

```
*****
```

```
*****
```

```
*****
```

11. Zaimplementuj i przetestuj rekurencyjną funkcję `writePyramid()`. Argumentem funkcji powinien być znak, z którego budowana jest piramida i jej wysokość (liczba linii, z których jest zbudowana). Wskazówka: każdy z wierszy piramidy powinien być generowany za pomocą dwóch wywołań funkcji `writeLine()`. Pierwsze

wywołanie generuje spację, drugie wywołanie generuje linię znaków. W poniższym przykładzie wywołania funkcji `writePyramid(3, '*')` znak `_` reprezentuje spację.

```

_ _ _ *
_ _ ***
_ _*****

```

12. W tym zadaniu obliczamy na kilka sposobów wartość x^n dla pewnego $n \geq 0$.

- Napisz iteracyjną funkcję `power1` obliczającą wartość x^n .
- Napisz rekurencyjną funkcję `power2` obliczającą wartość x^n za pomocą następującej rekurencyjnej definicji:

$$x^n = \begin{cases} 1 & \text{dla } n = 0 \\ x \cdot x^{n-1} & \text{dla } n > 0. \end{cases}$$

- Napisz rekurencyjną funkcję `power3` obliczającą wartość x^n za pomocą następującej rekurencyjnej definicji:

$$x^n = \begin{cases} 1 & \text{dla } n = 0 \\ (x^{n/2})^2 & \text{dla } n \text{ parzystego} \\ x \cdot (x^{n/2})^2 & \text{dla } n \text{ nieparzystego.} \end{cases}$$

Ile mnożeń wykonają funkcje `power1`, `power2`, `power3` obliczając 3^{32} ? 3^{19} ?