

## Lista\_1

## Zad. 1

Napisz programy zwracający  $n$ -ty wyraz ciągu Fibonacciego  $F(n)$  dwoma sposobami.

Oblicz  $F(n)$  według algorytmów zapisanym w „pseudokodzie”:

1. algorytm wielomianowy

```
funkcja fib1(n)
  if n=0 return 0
  utwórz tablicę F[0...n]
  F[0]=0, F[1]=1
  for i=2 to n:
    F[i]=F[i-1]+F[i-2];
  return F[n];
```

2. z definicji rekurencyjnej

```
funkcja fib1(n)
  if n=0: return 0
  if n=1: return 1
  return fib1(n-1) + fib1(n-2)
```

## Zad. 2

Wykonaj to samo co w zad.1 dla funkcji silnia –  $n!$

## Zad. 3

Zbuduj strukturę umożliwiającą pomiar czasu wykonywania kodu skorzystaj z *functional global variable*. Sprawdź, który ze sposobów z zad. 1. i 2 jest szybszy.

## Zad. 4

W LabVIEW można wykorzystywać *formula node* wykorzystaj ją przy zadaniu 1 i 2 (przypadki bez użycia rekurencji). Zbadaj wpływ tej formuły na szybkość wykonywania kodu. Czy zastosowanie jej ma znaczenie w porównaniu do kodu napisanego graficznie?