**Szybkie potęgowanie modulo n i Małe twierdzenie Fermata**

1. Obliczyć ręcznie
	1. $258^{1758} mod 13$ - wykorzystaj algorytm szybkiego potęgowania
	2. $258^{1758} mod 13$ - wykorzystaj Małe Twierdzenie Fermata (a potem ewentualnie algorytm szybkiego

**Do zadań 2-3:**

WE: całkowite liczby dodatnie $a,k,n$

WY: liczba $a^{k} mod n$

1. Napisać algorytm naiwny, który oblicza potęgę modulo n wykonując kolejne mnożenia i redukując reszty.
2. Napisać algorytm wykonujący szybkie potęgowanie modulo n.

 W pierwszym kroku zredukować resztę a modulo n

Wskazówka: wykorzystać zmienne pomocnicze

r – przechowująca aktualną resztę z dzielenia przez 2

q – przechowująca aktualny iloraz z dzielenia przez 2

s – przechowująca aktualną wartość iloczynu mod n

kw – przechowująca aktualny kwadrat mod n

1. Napisać algorytm redukujący wykładnik potęgi wykorzystując małe twierdzenie Fermata, a następnie obliczyc potęgę korzystając z algorytmu szybkiego potęgowania.

WE: całkowite liczby dodatnie $a,k,p$ , p- musi być liczbą pierwszą

WY: $a^{k} mod p$

Kolejne kroki:

redukcja k modulo p-1

zastosowanie algorytmu z zadania 3 do obliczenia potęgi

1. Do algorytmu z zadania 4. Dodać sprawdzanie, czy p jest liczbą pierwszą algorytmem naiwnym tzn. wykonując dzielenia przez kolejne liczby od 2 do pierwiastek z p.