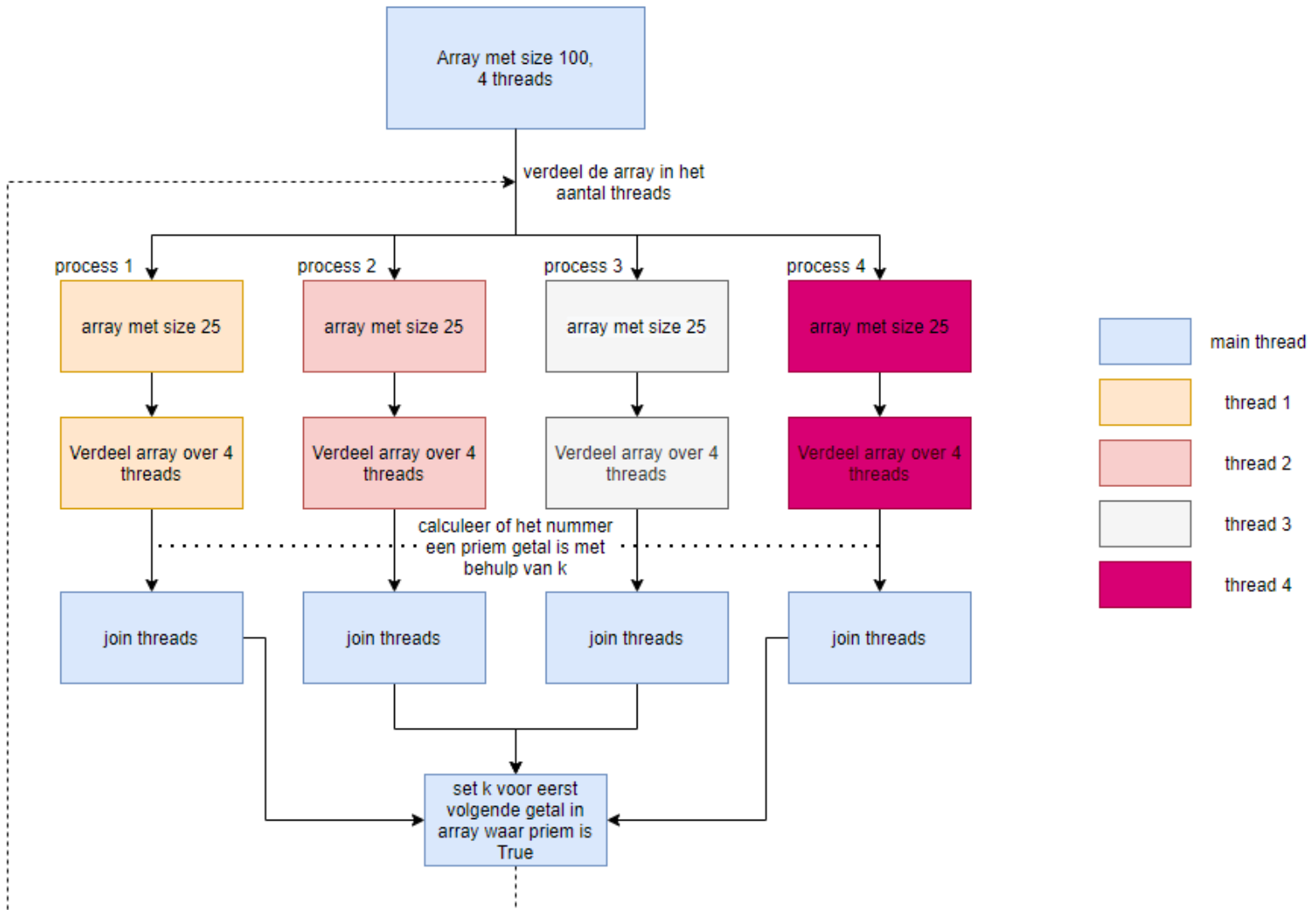
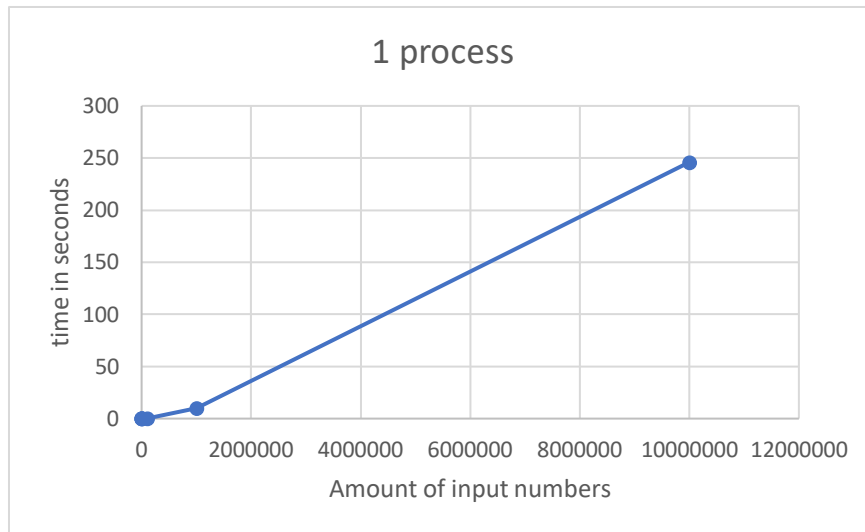


# Analyse Zeef van Eratosthenes.

Vanwegen het niet handig plannen en het niet uitkomen van het VM-gedeelte van deze opdracht heb ik alleen mijn code kunnen testen zonder multi processing. Wel heb ik een concept gemaakt om te laten zien hoe ik het zou aanpakken met meerdere threads.





getallen input	tijd in seconden
10	0
100	0
1000	0.00098
10000	0.024
100000	0.39
1000000	9.77
10,000,000	246

In de grafieken hierboven kun je zien dat je met een constante toename in het aantal input getallen, je een toenemende stijgende lijn zult zien. Dit komt door mijn while loop die voor ieder getal in de lijst een for loop moet doen door de hele lijst. In het geval van een lijst van 100 krijg je dus 100 x 100 getallen die hij vergelijkt.

Door het verdelen over bijvoorbeeld 4 threads zou het bij een hoog aantal input getallen zeker heel veel tijd moeten schelen.

Bij heel weinig input getallen duurt het verdelen over de threads waarschijnlijk langer dan het proces zelf als je het met 1 thread zou doen.

Vanwege een beperking aan goede hardware heb ik niet verder kunnen testen dan 10 miljoen input getallen.

#### Reflectie:

Ik vond het leuk om andere opdrachten te leren multi-threaden in linux, echter had ik hier wel wat moeite mee. Ik vind het dus jammer dat bij deze opdracht het niet gelukt is dit voor elkaar te krijgen maar ik heb wel weer nieuwe dingen geleerd. De code schrijven om de priemgetallen te vinden op basis van de pseudocode in de opdracht had ik dan weer wat minder moeite mee, ik hoop hier dan ook de meeste punten mee binnen te slepen.

Puntje bij paaltje vind ik dat ik niet helemaal geslaagd ben deze opdracht te voltooien maar de basis wel neer heb gezet voor verdere mogelijkheden.