O1i

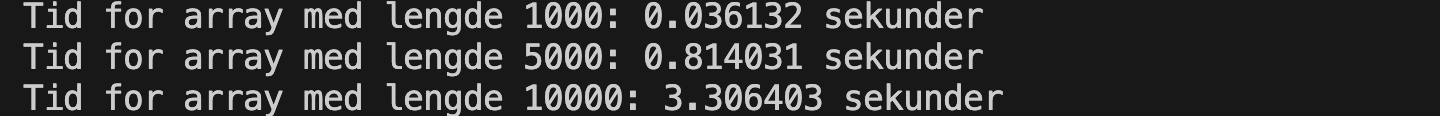
**Algoritmen**

Algoritmen tar inn en array som parameter med prisendringene for hver dag. Den bruker en dobbel for-løkke. Den ytterste løkken itererer over alle de potensielle kjøps-dagene, imens den innerste løkken itererer over alle dagene som kommer etter den nåværende kjøps dagen. Algoritmen holder styr på den nåværende prisen og profitten løpende, og gjør en sammenlignings-operasjon i form av en if-løkke inne i den innerste for-løkken. Hvis den nåværende profitten viser seg å være større enn profitten lagret som *maks\_profitt,* så oppdateres både *maks\_profitt* samt hvilke kjøp og salgs-dager dette er.

**Tidskompleksitet**

Hypotesen min er at algoritmen har en tidskompleksitet på O(n2) på grunn av den doble for-løkken. Nå er det jo sånn at den innerste for-løkken kun sjekker dagene etter den nåværende kjøps-dagen så jeg tror ikke nødvendigvis at den er nøyaktig O(n2).

**Tidtaking**



Dette var resultatene av tidtakingen. Som man ser så tar arrayen med lengde 10000 omtrent 100 ganger så lang tid som den med 1000, dette stemmer overens med O(n2).

Når datasetter 5 dobles (1000-5000) så burde tiden 25 dobles fordi (52 = 25).

A screenshot of a phone

Description automatically generated

Som sett på utregningen stemmer dette ganske mye overens.