Analisi ordinamento vettori:

# Programmi creati:

1. Bubblesort 1
2. Bubblesort 2
3. Bubblesort 3
4. Insertion 1
5. Insertion 2
6. Insertion 3
7. Selection 1
8. Selection 2
9. Selection 3
10. Ordinamento di complessità lineare.

# Bubblesort:

## Bubblesort 1:

Doppio ciclo for, che aumenta il numero di operazioni per un valore quadratico in base al numero di numeri da ordinare, quindi esponenzialmente.

Consiste nel leggere tutti i numeri fino a quando non sono tutti ordinati, uno successivamente all’altro, ed eventualmente scambiandoli se il secondo è più piccolo del primo letto.

## Bubblesort 2:

Simile al primo, ma ottimizzando il numero di cicli da rieseguire ogni volta.

## Bubblesort 3:

Simile a secondo, ma che controlla anche se il vettore è completamente ordinato, e se lo è si ferma in anticipo.

# Insertion:

## Insertion 1:

Algoritmo migliore del Bubblesort. Cerca il valore maggiore e lo sposta in fondo scambiando il valore presente, e diminuisce il numero di cicli da fare ogni volta che esegue l’operazione (inizia con n cicli, e ogni volta che ne esegue uno diminuisce di 1).

## Insertion 2:

Simile al primo, solo che cerca anche il valore minore scambiandolo con quello all’inizio.  
Si aggiunge il controllo del maggiore e minore trovato, e nel caso risultino tutti ordinati (i risultati sono insoddisfacenti), si ferma in anticipo.

## Insertion 3:

Ulteriori ottimizzazioni sul secondo, con modifiche minori alle condizioni e scambi, evita ulteriormente i cicli inutili.

# Selection:

## Selection 1:

Utilizzando 2 vettori, ordina i valori cercando il numero minore, sostituendolo in quello originale con un valore bandiera da ignorare, e aggiungendo il numero minore trovato al vettore finale.

## Selection 2:

Simile al primo ma “comprime” il primo vettore per non rieseguire i cicli in modo inutile per valori già ordinati; quindi, sposta i valori bandiera in fondo.

## Selection 3:

Simile all’Insertion 2, e un misto con il Selection 2, cerca sia maggiore che minore per spostarli in fondo o all’inizio del vettore finale, e comprime quello originale per non ricalcolare valori già fatti, controlla anche se tutti i valori sono ordinati e nel caso si ferma in anticipo.

# Tempi:

I tempi di tutti gli algoritmi sono segnati sul foglio Excel pubblico su classroom.

## Osservazioni:

Gli algoritmi sono disposti in ordine tramite un menù, i più veloci sono solitamente gli Insertion 1,2,3, specialmente il 2 e 3.  
I più lenti sono i Bubblesort.

Il Selection 3 è molto simile per funzionamento all’Insertion 2, e infatti ha tempi simili.

# Ordinamento di complessità lineare:

Creo un indice (vettore che rappresenta i numeri) e in una sola lettura dell’intero vettore da ordinare, conto quante volte il numero si ripete e conto quante volte un numero si ripete e lo aumento nell’indice, poi creo un vettore finale in cui inserisco tutti i numeri dell’indice per il numero di volte che un numero si ripete.

## ANALISI:

Un ciclo for conta i numeri dal minimo al massimo e incrementa l’indice, una volta fatto crea un vettore finale in base ai risultati dell’indice con i numeri, usando altri cicli for che vengono eseguiti per un numero di volte pari a quelle presenti nel vettore iniziale.