



K-DA LIBRARY

Review



0	1	2	3	4	5	6	7	8	9

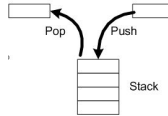


Problems of the previous presentation

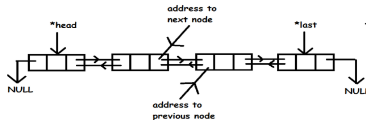
- Kinect-data
- Speed analysis

Data structures to implement

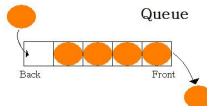
- Stack



- Double linked list



- Queue

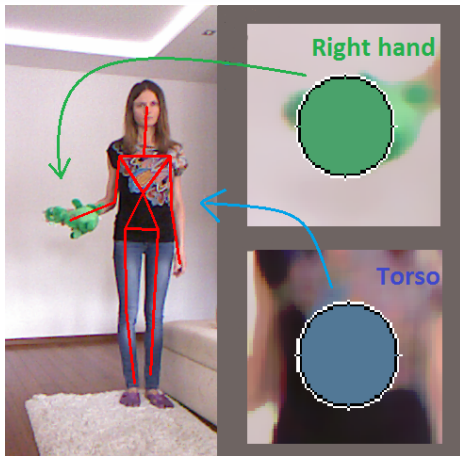
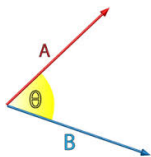


Complexity Analysis

Data Estructure	Storage	Using the methods
Stack	$O(n)$	$O(n)$
Linked list	$O(1)$	$O(n)$
Queue	$O(1)$	$O(n)$

Class conversion

- `convertir(string pjoint1, string pjoint2, int n);`
- `llenarArregloAngulos();`
- `getArregloAngulos();`



$$\vec{A} \cdot \vec{B} = A B \cos \theta$$

Class compara

- `sacapromedios(double arreglo);`
- `arreglo_promedio(double arreglo_prom1, double arreglo_prom2);`

sacapromedios

Array recibido:

$[n, k, \dots, l, m]$

Array retornado:

$[prom(n, k, \dots, l, m)]$

arreglo_promedio

Array recibido:

$[prom(n1, k1, \dots, l1, m1)]$

$[prom(n2, k2, \dots, l2, m2)]$

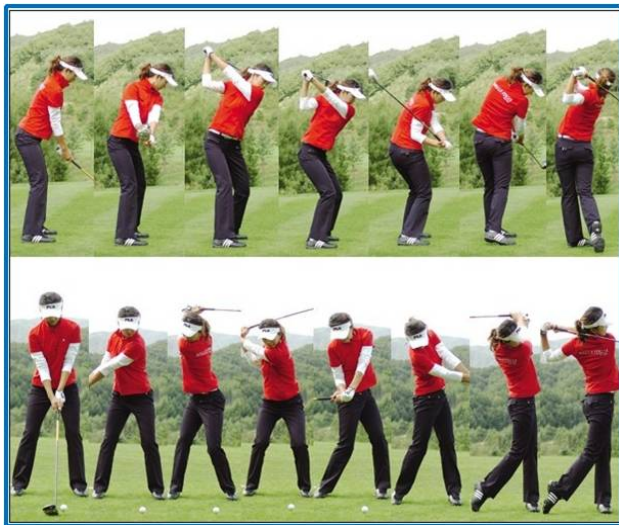
Array retornado:

$[1, 0, 0, 1, 0, 1, 1, 0, 1, 1]$

Class compara

- `comparar_angulos(int promedio);`
- `comparar_velocidad(int pSizeMov1, int pSizeMov2);`





Gracias