Ordenamiento

```
void ordenar_por_seleccion(int vector[MAX], int tope){
    int auxiliar;
    int pos_minimo;
    for(int i = 0; i < (tope - 1); i++) {
        pos_minimo = i;
        for (int j = i + 1; j < tope; j++) {
            if (vector[pos_minimo] > vector [j]) {
                pos_minimo = j;
            }
        }
        auxiliar = vector [i];
        vector [i] = vector [pos_minimo];
        vector [pos_minimo] = auxiliar;
    }
}
```

```
void ordenar_por_insercion(int vector[MAX] , int tope ){
    int j, auxiliar ;
    for (int i = 1; i < tope ; i++) {
        j = i;
        auxiliar = vector[i];

        while((j > 0) && (auxiliar < vector[j - 1])){
            vector[j] = vector[j - 1];
            j--;
        }
        vector[j] = auxiliar;
    }
}
```

```
void ordenar_por_burbujeo (int vector[MAX], int tope){
    int auxiliar;

    for (int i = 0; i < tope; i++) {
        for (int j = 0; j < (tope - i - 1); j++) {
            if(vector[j] > vector[j + 1]) {
                auxiliar = vector [j];
                vector [j] = vector [j + 1];
                vector [j + 1] = auxiliar ;
            }
        }
    }
}
```

```
void ordenar_por_burbujeo_mejorado (int vector[MAX], int tope){
    int j = 0, i, aux;
    bool esta_ordenado = false;

    while((j < tope) && (!esta_ordenado)){
        esta_ordenado = true ;
        for (i = 0; i < tope - 1; i++) {
            if(vector[i] > vector[i+1]) {
                aux = vector[i];
                vector[i] = vector[i + 1];
                vector[i + 1] = aux ;
                esta_ordenado = false ;
            }
        }
        j++;
    }
}
```