GSI064 – Resolução de Problemas
João Henrique

```
//Exemplo de qsort
#include <stdio.h>
#include <stdlib.h>

int compare (const void * a, const void * b){
  return ( *(int*)a - *(int*)b );
}

main(){
  int i, N, X[10000]={0};

  scanf("%d", &N);
  for(i=0; i<N; i++)
        scanf("%d", &X[i]);

  qsort(X, N, sizeof(int), compare);

  printf ("\nNumeros Ordenados:\n");
  for(i=0; i<N; i++)
        printf ("%d\n", X[i]);
}
```

```
// Exemplo de ordenação em C++
#include <iostream>     // std::cout
#include <algorithm>    // std::sort
#include <vector>       // std::vector

bool myfunction (int i,int j) { return (i<j); }

struct myclass {
  bool operator() (int i,int j) { return (i<j);}
} myobject;

int main () {
  int myints[] = {32,71,12,45,26,80,53,33};
  std::vector<int> myvector (myints, myints+8);               // 32 71 12 45 26 80 53 33

  // using default comparison (operator <):
  std::sort (myvector.begin(), myvector.begin()+4);           //(12 32 45 71)26 80 53 33

  // using function as comp
  std::sort (myvector.begin()+4, myvector.end(), myfunction); // 12 32 45 71(26 33 53 80)

  // using object as comp
  std::sort (myvector.begin(), myvector.end(), myobject);     //(12 26 32 33 45 53 71 80)

  // print out content:
  std::cout << "myvector contains:";
  for (std::vector<int>::iterator it=myvector.begin(); it!=myvector.end(); ++it)
    std::cout << ' ' << *it;
  std::cout << '\n';

  return 0;
}
```

```
//Exemplo de Contador Ordenado (Counting Sort)
#include <stdio.h>

main(){
 int N, Y, X[101]={0}, i;

 scanf("%d", &N);
 while(N--) {
        scanf("%d", &Y);
        X[Y]++;
   }
 for(i=0; i<101; i++){
        if(X[i] != 0)   printf("Contador de %d = %d\n", i, X[i]);
 }
}
```