```
Script started on 2022-12-08 14:39:45-06:00 [TERM="xterm" TTY="/dev/pts/8" COLUMNS=
j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ pwd
/home/students/j_pec2/JPMainDir/CSC122/Port3/TempSortLab
j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ cat TempSortInfo.txt
Jack Pec

CSC122-001

Template Sort Lab

EXTRA CREDIT

Overall level 2


Desc:

It's the "I want it to go that way"
labj_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ show-code TempSortDriver.cpp


TempSortDriver.cpp:
```

```cpp
#include <iostream>
//#include <cmath>
#include <vector>
#include <string>
#include <cstring>

using namespace std;

//////////////////start of template stuff

template <typename ItemType>
void swap(ItemType & x, ItemType & y)
{
    ItemType t = x;
    x = y;
    y = t;
    return;
}

template <size_t N>
void swap(char (&a)[N], char (&b)[N])
{
    cerr<<"my non-type template swap...'"<<a
        <<"' ("<<sizeof(a)<<") & '"<<b<<"' ("<<sizeof(b)<<")\n";
    char c[N];
    strcpy(c, a);
    strcpy(a, b);
    strcpy(b, c);
    return;
}



template <size_t A, size_t B>
struct Max_of
{
    enum { value = (A > B ? A : B) };
};

template <size_t N, size_t M>
void swap(char (&a)[N], char (&b)[M])
{
    cerr<<"my non-type template swap...'"<<
        a<<"' ("<<sizeof(a)<<") & '"<<b
        <<"' ("<<sizeof(b)<<")\n";
    char c[Max_of<N,M>::value];
    strcpy(c, a);
    strncpy(a, b, N-1);
    a[N-1] = '\0';
    strncpy(b, c, M-1);
    b[M-1] = '\0';
    return;
}

//// end of swap stuff

template <typename ArrayType, typename CompareType>
void sort1(ArrayType & array, const size_t & low,
           const size_t & high,//inc
           const CompareType & f )
{
    size_t done_thru, position;
    bool did_swap;
    did_swap = true;
    done_thru = high;
    while ((done_thru >= low) && (did_swap))
    {
        did_swap = false;
        for (position = low; position < done_thru; position++)
        {
            if (f(array[position],array[position+1]))
            {
                ::swap(array[position], array[position+1]);
                did_swap = true;
            }
        }
        done_thru--;
    }
    return;
}


////start of compare functions
template <typename T>
```

```cpp
bool ascend(const T& a,const T& b)
{

    bool returnVal = false;

    if(a > b)
    {
        returnVal = true;
    }
    return returnVal;


}

template <typename T>
bool descend(const T& a,const T& b)
{

    bool returnVal = false;

    if(a < b)
    {
        returnVal = true;
    }
    return returnVal;


}


template <size_t N>
bool ascend(const char(&a)[N],const char(&b)[N])
{

    bool returnVal = false;

    if(std::strcmp(a,b) > 0)
    {
        returnVal = true;
    }
    return returnVal;


}

template <size_t N>
bool descend(const char(&a)[N],const char(&b)[N])
{

    bool returnVal = false;

    if(std::strcmp(a,b) < 0)
    {
        returnVal = true;
    }
    return returnVal;


}

////////////////////////////////////end of template stuff


int main(void)
{

    short t[5] = {1,2,7,4,6};

    double a[5] = {1.6,2.9,3.1,4.2,6.4};

    char b [5] = {'a','b','c','d','g'};


    std::vector<std::string> colour1 = { "Blue",
                                         "Red",
                                         "Orange",
                                         "Yellow"
                                       };

    //array of c_strs
    const size_t colour3len = 4;


    char colour3[colour3len][8] = { "Blue2",
                                    "Red2",
                                    "Orange2",
                                    "Yellow2"
                                  };

    sort1(a,static_cast<size_t>(0),
          static_cast<size_t>(4),
          ascend<double>);
    sort1(b,static_cast<size_t>(0),
          static_cast<size_t>(4),
          ascend<char>);

    sort1(t,static_cast<size_t>(0),
          static_cast<size_t>(4),
          ascend<short>);

    sort1(colour1,static_cast<size_t>(0),
          static_cast<size_t>(3),
          ascend<std::string>);

    sort1(colour3,static_cast<size_t>(0),
          static_cast<size_t>(3),
          ascend<8>);

```

```
193
194     //sort1(c,static_cast<size_t>(0),clen, ascend);
195
196     for(size_t i = 0; i < 5; i++)
197     {
198         std::cout << a[i] <<" ";
199     }
200
201     std::cout << "\n";
202
203     for(size_t i = 0; i < 5; i++)
204     {
205         std::cout << t[i] <<" ";
206     }
207
208     std::cout << "\n";
209
210     for(size_t i = 0; i < 5; i++)
211     {
212         std::cout << b[i] << " ";
213     }
214
215     std::cout << "\n";
216     std::cout << "\n";
217
218     for(size_t i = 0; i < 4; i++)
219     {
220         std::cout << colour1[i];
221         std::cout << "\n";
222     }
223
224     std::cout << "\n";
225
226
227
228     for(size_t i = 0; i < 4; i++)
229     {
230         std::cout << colour3[i];
231         std::cout << "\n";
232     }
233
234     std::cout << "\n";
235
236
237     sort1(a,static_cast<size_t>(0),
238           static_cast<size_t>(5),
239           descend<double>);
240
241     sort1(b,static_cast<size_t>(0),
242           static_cast<size_t>(5),
243           descend<char>);
244
245     sort1(t,static_cast<size_t>(0),
246           static_cast<size_t>(4),
247           descend<short>);
248
249     sort1(colour1,static_cast<size_t>(0),
250           static_cast<size_t>(3),
251           descend<std::string>);
252
253
254     sort1(colour3,static_cast<size_t>(0),
255           static_cast<size_t>(3),
256           descend<8>);
257
258
259     for(size_t i = 0; i < 5; i++)
260     {
261         std::cout << a[i] << " ";
262     }
263
264     std::cout << "\n";
265
266     for(size_t i = 0; i < 5; i++)
267     {
268         std::cout << t[i] <<" ";
269     }
270
271     std::cout << "\n";
272
273     for(size_t i = 0; i < 5; i++)
274     {
275         std::cout << b[i] << " ";
276     }
277
278     std::cout << "\n";
279
280
281     std::cout << "\n";
282
283     for(size_t i = 0; i < 4; i++)
284     {
285         std::cout << colour1[i];
286         std::cout << "\n";
287     }
288
289     std::cout << "\n";
290
291     for(size_t i = 0; i < 4; i++)
292     {
293         std::cout << colour3[i];
294         std::cout << "\n";
295     }
296
297
298     return 0;
299 }
j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ CPP TempSortDriver.cpp
```

```
TempSortDriver.cpp***


j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ ./TempSortDriver.out
my non-type template swap...'Red2' (8) & 'Orange2' (8)
1.6 2.9 3.1 4.2 6.4
1 2 4 6 7
a b c d g

Blue
Orange
Red
Yellow

Blue2
Orange2
Red2
Yellow2

my non-type template swap...'Blue2' (8) & 'Orange2' (8)
my non-type template swap...'Blue2' (8) & 'Red2' (8)
my non-type template swap...'Blue2' (8) & 'Yellow2' (8)
my non-type template swap...'Orange2' (8) & 'Red2' (8)
my non-type template swap...'Orange2' (8) & 'Yellow2' (8)
my non-type template swap...'Red2' (8) & 'Yellow2' (8)
6.4 4.2 3.1 2.9 1.6
7 6 4 2 1
g d c b a

Yellow
Red
Orange
Blue

Yellow2
Red2
Orange2
Blue2
j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ cat TempSortTPQ.txt
1. What things might cause your new sort template to fail to instantiate?

Lot's of things, mainly bad syntax with the instantiation.


2. Where should the overloaded swap form go?
(In the library or the main application?)
If in the library, should it be in the header
 with the swap template or in the implementation file?
Does it matter to the compiler where it is? Why/Why not?

I'm keeping in the main app, having templates in
header files is nasty business and requires alot of
time and effort to do, which I'm not gonna do
on a lab worth only 2 levels


3. Did you need to make any changes to your original swap template?

Yes, I made it so it takes in functions for the comparisons.


4. Which comparisons did you write as plain functions?
Function objects? Were any of them templated?
Could/Should they have been?
(Hint: you should use at least one plain function
and one function object class to show you can do both.)

None of the comparisons there plain functions, but I
could use them if I wanted to, all of them are
templated function objects passed through to the sort
function



j_pec2@ares:~/JPMainDir/CSC122/Port3/TempSortLab$ exit
exit

Script done on 2022-12-08 14:41:26-06:00 [COMMAND_EXIT_CODE="0"]
```