Hi, congratulation guys! I believe you all doing good for midterm and enjoy the break. Now let’s get a little idea about your assignment 4 before you go on a trip ( Is it due this weekend? Or next weekend.)

Here is just an example for the function “merge\_sort” which professor ask for. Generally it’s not easy to explain by words but you may try to read the code. I will add more comments later to help you understand. Once you read the code, you can try to write your own code for it and feel free to email me if you have any question during the break. I may not be able to meet but I can reply your questions by mail everyday.

You all do a really good job this semester and learn more than me when I took 121 class. I am so proud of you. It’s time to enjoy the break !

#include <iostream>

#include <vector>

using namespace std;

void print\_vector(const vector<int> &v)

{

unsigned long n = v.size();

for(int i=0; i<n; i++)

{

cout << v[i] << " ";

}

cout << endl;

}

vector<int> merge\_sort(const vector<int> &v)

{

if(v.size() == 1) return v;

else{

vector<int> sorted;

unsigned long mid = v.size()/2;

vector<int> left(v.begin(), v.begin()+mid);

vector<int> right(v.begin() + mid, v.end());

left = merge\_sort(left); //print\_vector(left);

right = merge\_sort(right); //print\_vector(right);

vector<int>::iterator v\_left = left.begin();

vector<int>::iterator v\_right = right.begin();

while(v\_left != left.end() && v\_right != right.end())

{

if(\*v\_left < \*v\_right)

{

sorted.push\_back(\*v\_left);

v\_left++;

}

else

{

sorted.push\_back(\*v\_right);

v\_right++;

}

}

while(v\_left != left.end())

{

sorted.push\_back(\*v\_left);

v\_left++;

}

while(v\_right != right.end())

{

sorted.push\_back(\*v\_right);

v\_right++;

}

return sorted;

}

}

int main(int argc, const char \* argv[]) {

vector<int> v = {8,2,3,4,5,6,7,1};

vector<int> receiver = merge\_sort(v);

print\_vector(receiver);

return 0;

}