I have finished the project. I did not use banned STL. I don’t have bugs that I haven’t fixed.

Here are my test cases except MatchMaker class

void testPersonProfile() {

PersonProfile pp("allen", "allenwang2333@gmail.com");

AttValPair atv1("hobby", "sleeping");

AttValPair atv2("hobby", "eating");

AttValPair atv3("hobby", "playing guitar");

AttValPair atv4("trait", "chill");

AttValPair atv5("trait", "patient");

AttValPair atv6("trait", "chill"); *// test repetition*

pp.AddAttValPair(atv1);

pp.AddAttValPair(atv2);

pp.AddAttValPair(atv3);

pp.AddAttValPair(atv4);

pp.AddAttValPair(atv5);

pp.AddAttValPair(atv6);

AttValPair a;

vector<AttValPair> va;

for (int i = 0; i < pp.GetNumAttValPairs(); i++) {

if (pp.GetAttVal(i, a)) {

va.push\_back(a);

}

}

assert(pp.GetNumAttValPairs() == 5);

assert(pp.GetName() == "allen");

assert(pp.GetEmail() == "allenwang2333@gmail.com");

assert(va.size() == 5);

assert(va.at(0) == atv1);

assert(va.at(1) == atv2);

assert(va.at(2) == atv3);

assert(va.at(3) == atv4);

assert(va.at(4) == atv5);

assert(!pp.GetAttVal(7, atv1));

assert(!pp.GetAttVal(-1, atv1));

assert(atv1 == AttValPair("hobby", "sleeping"));

cerr << "passed PersonProfile test" << endl;

}

void testAttributeTranslator() {

AttributeTranslator at;

assert(!at.Load("heiheihie.docv"));

assert(at.Load("translator.txt"));

vector<AttValPair> av1 = at.FindCompatibleAttValPairs(AttValPair("lalalalalala", "nihao"));

assert(av1.size() == 0);

vector<AttValPair> av2 = at.FindCompatibleAttValPairs(AttValPair("job", "chef"));

assert(av2.size() == 17);

vector<AttValPair> av3 = at.FindCompatibleAttValPairs(AttValPair("job", "editor"));

assert(av3.size() == 15); *// test repeated case;*

cerr << "passed AttributeTranslator test" << endl;

}

void testMemberDatabase() {

MemberDatabase db;

assert(!db.LoadDatabase("heiheiehi.duhd"));

assert(db.LoadDatabase("members.txt"));

const PersonProfile\* ppp1 = db.GetMemberByEmail("allenwang2333@gmail.com");

assert(ppp1 == nullptr);

const PersonProfile\* ppp2 = db.GetMemberByEmail("MHa7@sky.com");

assert(ppp2->GetName() == "Mordechai Hayes");

assert(ppp2->GetEmail() == "MHa7@sky.com");

assert(ppp2->GetNumAttValPairs() == 6);

const PersonProfile\* ppp3 = db.GetMemberByEmail("AbFow2483@charter.net");

assert(ppp3->GetName() == "Abdullah Fowler");

assert(ppp3->GetEmail() == "AbFow2483@charter.net");

assert(ppp3->GetNumAttValPairs() == 12);

const PersonProfile\* ppp4 = db.GetMemberByEmail("OlSkinn@yahoo.com");

assert(ppp4->GetName() == "Oliver Skinner");

assert(ppp4->GetEmail() == "OlSkinn@yahoo.com");

assert(ppp4->GetNumAttValPairs() == 12 );

AttValPair av1("trait", "prejudiced");

vector<std::string> vs1 = db.FindMatchingMembers(av1);

assert(vs1.size() == 1394);

AttValPair av2("hobby", "cosplaying");

vector<std::string> vs2 = db.FindMatchingMembers(av2);

assert(vs2.size() == 6257);

AttValPair av3("job", "psychologist");

vector<std::string> vs3 = db.FindMatchingMembers(av3);

assert(vs3.size() == 4336);

AttValPair av4("hobby", "painting");

vector<std::string> vs4 = db.FindMatchingMembers(av4);

assert(vs4.size() == 6280);

AttValPair av5("eoahfe", "diuh");

vector<std::string> vs5 = db.FindMatchingMembers(av5);

assert(vs5.empty());

cerr << "passed MemberDatabase test" << endl;

}

void testRadixTree() {

RadixTree<int> tree;

tree.insert("allen", 1);

tree.insert("phil", 2);

tree.insert("apple", 3);

tree.insert("applepencil", 4);

tree.insert("app", 5);

tree.insert("all", 6);

tree.insert("all", 7);

tree.insert("box", 8);

assert(\*(tree.search("allen")) == 1);

assert(\*(tree.search("phil")) == 2);

assert(\*(tree.search("apple")) == 3);

assert(\*(tree.search("applepencil")) == 4);

assert(\*(tree.search("app")) == 5);

assert(\*(tree.search("all")) == 7);

assert(\*(tree.search("box")) == 8);

assert(tree.search("") == nullptr);

assert(tree.search("aaaaaaa") == nullptr);

cerr << "passed RadixT

For MatchMaker class, I compared output of 10 people according to the sample program and found that the results are the same. So, I didn’t write separate test cases for it.