```
Script started on 2021-04-24 18:30:47-0500
m_sadaf1@ares:~$ pwd
/home/students/m_sadaf1
m_sadaf1@ares:~$ cat traits.cpp
#include "traits.h"
#include <fstream>
#include <iostream>
#include <limits>
#include <string>
#include <sstream>

using namespace std;

inline bool file_exist(const string & name)
{
    bool ret;
    ifstream file;
    file.open(name);
    if (!file)
    {
        ret = false;
    }
    else
    {
        ret = true;
    }
    file.close();
    file.clear();
    return ret;
}

int main(void)
{
    ifstream input;
    ofstream output;
    string filename;

    cout << "Enter the name of the input file: ";
    getline(cin, filename);
    input.open(filename);
    while (!input)
    {
        input.close();
        input.clear();

        cout << "File does not exist.\n Enter file name: ";
        getline(cin, filename);
        input.open(filename);
    }

    cout << filename << " selected as input file.";

    cout << "Enter a new output file name: ";
    getline(cin, filename);
    while (file_exist(filename))
    {
        cout << "Error" << filename << "already exists."
                "\nEnter a new file";
        getline(cin, filename);
    }
    output.open(filename);
    cout << filename << "selected as output file.";

    Experiment experiment;

    input.ws();
    while (!input.eof())
    {
        experiment.read(input);
        experiment.write(output);
        input.clear();
        input >> ws();
    }
    output.close();
    input.close();

    cout << "Hit ENTER to exit.";
    cin.ignore(numeric_limits<streamsize>::max(),
                           '\n');
    return 0;
}

m_sadaf1@ares:~$ cat traits.h
#ifndef TRAITS_EXPERIMENT_H
#define TRAITS_EXPERIMENT_H

#include <string>
#include <fstream>
#include <iostream>
#include <iomanip>
#include <cmath>
#include <limits>
```

```
class Experiment
{
    // Member Variables

    std::string exp_name;
    std::string creature;
    long population;
    std::string trt1_name;
    std::string trt2_name;
    std::string midtrt_name;
    long trt1_count;
    long trt2_count;

public:
        // Constructor

        Experiment(): exp_name(), creature(), population(),
        trt1_name(), trt2_name(), midtrt_name("Middle Trait"),
        trt1_count(), trt2_count() {}

        // Accessors // Files

        void read(std::istream& in)
        {
            using namespace std;

            in >> ws;

            // Population
            while(in.peek() == '#')
            {
                in.ignore(numeric_limits<streamsize>::max(),
                          '\n');
                in >> ws;
            }

            in >> population >> ws; // reads

            // Experiment Name
            while(in.peek() == '#')
            {
                in.ignore(numeric_limits<streamsize>::max(),
                          '\n');
                in >> ws;
            }

            getline(in, exp_name); //Reads name till newline

            size_t pos = exp_name.find('#');
                    // Find pos of first #

            if(pos != std::string::npos)
            {
                exp_name.erase(pos);
                // Erases everything from #
            }

            // Creature
            while (in.peek() == '#')
            {
                in.ignore(numeric_limits<streamsize>::max(),
                          '\n');
                in >> ws;
            }

            getline(in, creature);
            pos = creature.find('#');

            if(pos != std::string::npos)
            {
                creature.erase(pos);
            }

            // Trait 1
            while (in.peek() == '#')
            {
                in.ignore(numeric_limits<streamsize>::max(),
                          '\n');
                in >> ws;
            }

            getline(in, trt1_name);
            pos = trt1_name.find('#');

            if(pos != std::string::npos)
            {
                trt1_name.erase(pos);
            }
            // Trait 2
            while (in.peek() == '#')
            {
                in.ignore(numeric_limits<streamsize>::max(),
                          '\n');
                in >> ws;
            }
```

```
    getline(in, trt2_name);
    pos = trt2_name.find('#');

    if(pos != std::string::npos)
    {
        trt2_name.erase(pos);
    }

    // Trait 1 Count

    while (in.peek() == '#')
    {
        in.ignore(numeric_limits<streamsize>::max(),
                  '\n');
        in >> ws;
    }

    in >> trt1_count >> ws;

    // Trait 2 Count

    while (in.peek() == '#')
    {
        in.ignore(numeric_limits<streamsize>::max(),
                  '\n');
        in >> ws;
    }

}

void write(std::ostream& out) // Writes data to file
{
    using namespace std;

    // Find width of name

    string::size_type name_length = midtrt_name.length();

    if (trt1_name.length() > name_length)
    {
        name_length = trt1_name.length();
    }

    if (trt2_name.length() > name_length)
    {
        name_length = trt2_name.length();
    }

    // Find width of number

    short count_length;
    long midtrt_count = static_cast<double>
            (population - trt1_count -
            trt2_count);
    count_length = static_cast<short>
            (floor(log10(population) + 1));

    // Calculations

    double trt_prcnt1 = static_cast<double>
        (trt1_count) + static_cast<double>
        (midtrt_count) / 2.0) * 100
        / static_cast<double>(population);
    double trt_prcnt2 = static_cast<double>
        (trt2_count) + static_cast<double>
        (midtrt_count) / 2.0) * 100
        / static_cast<double>(population);

    short padding;
    short spacing = 4;

    short prcnt_length = 6;

    // Calc amount of space needed to center text
    padding = (80 - exp_name.length()) / 2;
    out << string(padding, ' ') << exp_name;
    padding = (80 - creature.length()) / 2;
    out << string(padding, ' ') << creature;

    // Trait 1

    out << right << setw(name_length) << trt1_name
            << ':' << string(spacing, ' ');
    out << right << setw(count_length) << trt1_count
            << ':' << string(spacing, ' ');
    out << right << setw(prcnt_length)
            << setprecision(2) << fixed
            << trt1_prcnt << "%\n";

    // Mixed Trait

    out << right << setw(name_length) << midtrt_name
            << ':' << string(spacing, ' ');
```

```
        out << right << setw(count_length)
                << midtrt_name;

        // Trait 2

        out << right << setw(name_length) << trt2_name
                << ':' << string((spacing-2, ' ')
                << '+' << ' ';
        out << right << setw(count_length) << trt2_count
                << string(spacing, ' ');
        out << right << setw(prcnt_length)
                << setprecision(2) << fixed
                << trt2_prcnt << "%\n";

        // Equals group under data

        out << right << string((name_length + spacing-1)
                , ' ') << string(count_length + 2, '-')
                << string(spacing, ' ')
                << string(7, '-');

        // Total

        out << right << setw(name_length) << "Total"
                << ':' << string(spacing, ' ');
        out << right << setw(count_length) << population
                << string(spacing, ' ');
        out << right << setw(prcnt_length)
                << setprecision(2) << fixed
                << trt1_prcnt + trt2_prcnt
                << "%\n\n\n";
    }
};

#endif /* TRAITS_EXPERIMENT_H */

m_sadaf1@ares:~$ CPP traits
traits.cpp***
In file included from traits.cpp:1:0:
traits.h: In member function ‘void
Experiment::write(std::ostream&)’:
traits.h:157:33: warning: conversion
to ‘long int’ from ‘double’ may
alter its value [-Wfloat-conversion]
          long midtrt_count = static_cast<double>
                              ^~~~~~~~~~~~~~~~~~
                  (population - trt1_count -
                   ~~~~~~~~~~~~~~~~~~~~~~~~
                   trt2_count);
                   ~~~~~~~~~~~
traits.h:167:37: error: expected
‘,’ or ‘;’ before
‘)’ token
                  (midtrt_count) / 2.0) * 100
                                      ^
traits.h:171:37: error: expected
‘,’ or ‘;’ before
‘)’ token
                  (midtrt_count) / 2.0) * 100
                                      ^
traits.h:180:48: warning:
conversion to ‘short int’ from
‘std::__cxx11::basic_string<char>::size_type
{aka long unsigned int}’ may alter its value
[-Wconversion]
             padding = (80 - exp_name.length()) / 2;
                       ~~~~~~~~~~~~~~~~~~~~~~^~~
traits.h:182:48: warning:
conversion to ‘short int’ from
‘std::__cxx11::basic_string<char>::size_type
{aka long unsigned int}’ may alter its value
[-Wconversion]
             padding = (80 - creature.length()) / 2;
                       ~~~~~~~~~~~~~~~~~~~~~~^~~
traits.h:187:45: warning:
conversion to ‘int’ from
‘std::__cxx11::basic_string<char>::size_type
{aka long unsigned int}’ may alter its value
[-Wconversion]
             out << right << setw(name_length) <<
             trt1_name
                                             ^
traits.h:193:24: error:
‘trt1_prcnt’ was not declared in this scope
                    << trt1_prcnt << "%\n";
                       ^~~~~~~~~~
traits.h:193:24: note: suggested
alternative: ‘trt_prcnt2’
                    << trt1_prcnt << "%\n";
                       ^~~~~~~~~~
                       trt_prcnt2
traits.h:197:45: warning:
conversion to ‘int’ from
‘std::__cxx11::basic_string<char>::size_type
```

```
{aka long unsigned int}' may alter its value
[-Wconversion]
             out << right << setw(name_length) <<
             midtrt_name
                                               ^
traits.h:204:45: warning:
conversion to 'int' from
'std::__cxx11::basic_string<char>::size_type
{aka long unsigned int}' may alter its value
[-Wconversion]
             out << right << setw(name_length) <<
             trt2_name
                                               ^
traits.h:205:46: warning: left operand
of comma operator has no effect [-Wunused-value]
                     << ':' << string((spacing-2, ' ')
                                      ~~~~~~~^~
traits.h:206:24: warning: left shift
count >= width of type [-Wshift-count-overflow]
                     << '+' << ' ';
                        ^~~
traits.h:206:31: warning: left shift
count >= width of type [-Wshift-count-overflow]
                     << '+' << ' ';
                               ^~~
traits.h:205:37: error: expected
primary-expression before '(' token
                     << ':' << string((spacing-2, ' ')
                                     ^
traits.h:205:46: warning: left operand
of comma operator has no effect [-Wunused-value]
                     << ':' << string((spacing-2, ' ')
                                      ~~~~~~~^~
traits.h:206:24: warning: left shift
count >= width of type [-Wshift-count-overflow]
                     << '+' << ' ';
                        ^~~
traits.h:206:31: warning: left shift
count >= width of type [-Wshift-count-overflow]
                     << '+' << ' ';
                               ^~~
traits.h:206:34: error: expected
')' before ';' token
                     << '+' << ' ';
                                  ^
traits.h:211:24: error:
'trt2_prcnt' was not declared in this scope
                     << trt2_prcnt << "%\n";
                        ^~~~~~~~~~
traits.h:211:24: note: suggested
alternative: 'trt_prcnt2'
                     << trt2_prcnt << "%\n";
                        ^~~~~~~~~~
                        trt_prcnt2
traits.h:222:45: warning:
conversion to 'int' from
'std::__cxx11::basic_string<char>::size_type
{aka long unsigned int}' may alter its value
[-Wconversion]
             out << right << setw(name_length) <<
             "Total"                                  ^

traits.h:165:20: warning:
unused variable 'trt_prcnt1'
[-Wunused-variable]
             double trt_prcnt1 = static_cast<double>
                    ^~~~~~~~~~
traits.h:169:20: warning:
unused variable 'trt_prcnt2'
[-Wunused-variable]
             double trt_prcnt2 = static_cast<double>
                    ^~~~~~~~~~
traits.cpp: In function 'int main()':
traits.cpp:62:11: error:
'std::ifstream {aka class
std::basic_ifstream<char>}' has no member named
'ws'
     input.ws();
           ^~
traits.cpp:68:21: error: no matching
function for call to 'ws()'
         input >> ws();
                     ^
In file included from /usr/include/c++/7/istream:991:0,
                 from /usr/include/c++/7/fstream:38,
                 from traits.h:5,
                 from traits.cpp:1:
/usr/include/c++/7/bits/istream.tcc:1024:5:
note: candidate: template<class _CharT, class _Traits>
std::basic_istream<_CharT, _Traits>& std::ws(std::basic_istream<_CharT,
_Traits>&)
     ws(basic_istream<_CharT, _Traits>& __in)
     ^~
/usr/include/c++/7/bits/istream.tcc:1024:5:
```

```
note:   template argument deduction/substitution
failed:
traits.cpp:68:21: note:   candidate
expects 1 argument, 0 provided
         input >> ws();
                  ^



m_sadaf1@ares:~$ exit
exit

Script done on 2021-04-24 18:32:00-0500
```