```
Script started on 2023-11-26 13:57:48-06:00 [TERM="xterm" TTY="/dev/pts/0" COLUMNS=
ee43254@ares:~$ pwd
/home/students/ee43254
ee43254@ares:~$ cat pt_menu.info
Name: Kyle Enkhzul

Class: CSC121-001



Activity: Pick'n'Choose

Level: 6, 2 (base program), 1 (switch menu), 3 (functions)



Description:



This program allows a user to calculate the distance and midpoint between two

points using a menu system. They have three options to choose from. The first

option calculates the distance between two points. The second option calculates

the midpoint between two points. Finally, the third option quits the program.




ee43254@ares:~$ show-code pt_menu.cpp


pt_menu.cpp:
```

```cpp
#include <iostream>
#include <cmath>
#include <limits>

using namespace std;
constexpr streamsize INF_FLAG{numeric_limits<streamsize>::max()};

double calculateMidX(double x1, double x2) {
    return ((x1+x2)/2.0);
}

double calculateMidY(double y1, double y2) {
    return ((y1+y2)/2.0);
}

double calculateDistance(double x1, double x2, double y1, double y2) {
    return (sqrt(pow(x2-x1, 2) + pow(y2-y1, 2)));
}

void printDistance(double x1, double x2, double y1, double y2,
   double distance){
    cout << "\n(" << x1 << ", " << y1 << ") is " << distance
         << " units away from (" << x2 << ", " << y2 << ").\n\n";
}

void printMidpoint(double x1, double x2, double y1, double y2,
double midx, double midy) {
    cout << "\n The midpoint of the line segment from ("
         << x1 << ", " << y1 << ") and (" << x2 << ", " << y2
         << ")\n is (" << midx << ", " << midy << ").\n\n";
}

int main() {
    double x1, x2, y1, y2, midx, midy, distance;
    char garbage, choice;
    bool done = false;

    cout << "\t\t\t Welcome to the Point Menu Program!!!\t\n";

    while(!done) {
        cout << "\n\t1) Calculate Distance Between Two Points\n"
    "\t2) Calculate Midpoint of Two Points\n"
    "\t3) Quit\n\n"
    "\tChoice:    ";

        cin >> choice;
        cin.ignore(INF_FLAG, '\n');

        choice = static_cast<char>(toupper(choice));

        switch(choice) {
            case '1': case 'D':
    cout << "\nWhere is the first point? ";
    cin >> garbage >> x1 >> garbage >> y1 >> garbage;

    cout << "\nWhere is the second point? ";
    cin >> garbage >> x2 >> garbage >> y2 >> garbage;

    distance = calculateDistance(x1, x2, y1, y2);

    printDistance(x1, x2, y1, y2, distance);
    break;
            case '2': case 'M':
    cout << "\nWhere is the first point? ";
    cin >> garbage >> x1 >> garbage >> y1 >> garbage;

    cout << "\nWhere is the second point? ";
    cin >> garbage >> x2 >> garbage >> y2 >> garbage;

    midx = calculateMidX(x1, x2);
```

```
    71      midy = calculateMidY(y1, y2);
    72
    73      printMidpoint(x1, x2, y1, y2, midx, midy);
    74      break;
    75              case '3': case 'Q':
    76      cout << "\nThank you for using the PMP!!\n";
    77      cout << "\nHave a good day!\n";
    78      done = true;
    79      break;
    80              default:
    81      cout << "\nI'm sorry, that choice is invalid! \n";
    82      cout << "\nPlease try to read/type more carefully ";
    83      cout << "next time\n";
    84          }
    85      }
    86  }
    87
ee43254@ares:~$ CPP pt_menu
pt_menu.cpp***


ee43254@ares:~$ ./pt_menu.out
                        Welcome to the Point Menu Program!!!

        1) Calculate Distance Between Two Points
        2) Calculate Midpoint of Two Points
        3) Quit

        Choice:    n

I'm sorry, that choice is invalid!

Please try to read/type more carefully next time

        1) Calculate Distance Between Two Points
        2) Calculate Midpoint of Two Points
        3) Quit

        Choice:    mid

Where is the first point? (3.4, 12.2)

Where is the second point? (13.4, 12.2)

 The midpoint of the line segment from (3.4, 12.2) and (13.4, 12.2)
 is (8.4, 12.2).


        1) Calculate Distance Between Two Points
        2) Calculate Midpoint of Two Points
        3) Quit

        Choice:    D

Where is the first point? (3.4, 12.2)

Where is the second point? (13.4, 12.2)

(3.4, 12.2) is 10 units away from (13.4, 12.2).


        1) Calculate Distance Between Two Points
        2) Calculate Midpoint of Two Points
        3) Quit

        Choice:    3

Thank you for using the PMP!!

Have a good day!
ee43254@ares:~$ cat pt_menu.tpq
1. How do you get the menu to repeat until they choose the quit option?

        In order to get the menu to repeat until they've chosen the quit option is

        to use a helper bool variable. When they've chosen the quit option, the

        bool variable will be assigned to true which will make the while loop exit



2. How do you detect when they've entered something invalid?

        We can detect if they've entered something invalid by testing for only

        values we want. If we only accept the values we want, any other case will

        default to reprompting for a correct choice.



3. Can you easily check for both upper- and lower- case menu entries?

(What library function might help here?)

        We can easily check for both upper and lower case menu entries by using

        the toupper() function to test for only one case.



4. How can you dispose of the excess of a word they may have typed at the choice

prompt?

        We can dispose of the excess of a word they may have typed by using:

        constexpr streamsize INF_FLAG{numeric_limits<streamsize>::max()};
```

```
        cin.ignore(INF_FLAG, '\n');


5. How many tests would be needed to thoroughly (i.e. the menu, the
calculations, ...everything) test this application?
        Four tests would be needed to thoroughly test this application. It would
        need to test each of the options using either the number or first char.
        The other one would be to check if the quit option properly exits the loop
        and program. The last one would be any other input to see how the program
        handles invalid input.


6. Why is a switch preferred over a cascaded if for menu processing?
        A switch is mort efficient than a set of nested ifs. The ifs must go throug
        each and every comparison to narrow down the exact execution of code needed
        while switch expressions execute the necessary code directly. ee43254@ares
exit

Script done on 2023-11-26 13:59:26-06:00 [COMMAND_EXIT_CODE="0"]
```