```
Script started on 2023-09-25 15:29:16-05:00 [TERM="xterm" TTY="/dev/pts/2" COLUMNS=
ee43254@ares:~$ pwd
/home/students/ee43254
ee43254@ares:~$ cat midpoint.info
Name: Kyle Enkhzul
Class: CSC121-001

Activity: Point to Point
Level: 2.5, 1.5 (base program), 1 (point notation)

Description:

The user inputs two endpoints into the program. The program then takes the two
endpoints and calculates the midpoint between them.

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


midpoint.cpp:


#include <iostream>

using namespace std;

int main(void) {
        double x1, x2, y1, y2, midx, midy;
        char garbage;

        cout << "\t\n Welcome to the 2D Midpoint Program!!!\t\n";

        cout << "\n What is the first end-point? ";
        cin >> garbage >> x1 >> garbage >> y1 >> garbage;

        cout << "\n What is the second end-point? ";
        cin >> garbage >> x2 >> garbage >> y2 >> garbage;

        cout << "\n Thank you!!! Calculating... Done. \n";

        midx = (x1+x2)/2.0;
        midy = (y1+y2)/2.0;

        cout << "\n The midpoint of the line segment between ("
        << x1 << ", " << y1 << ") and (" << x2 << ", " << y2 << ")\n is ("
                        << midx << ", " << midy << ").\n";

        cout << "\nThank you for using the TMP!!\n";
        cout << "\nHave a good day\n";
}
ee43254@ares:~$ CPP midpoint
midpoint.cpp***


ee43254@ares:~$ ./midpoint.out

 Welcome to the 2D Midpoint Program!!!

 What is the first end-point? (3.4, 12.2)

 What is the second end-point? (13.4, 12.2)

 Thank you!!! Calculating... Done.

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

Thank you for using the TMP!!

Have a good day
ee43254@ares:~$ ./midpoint.out

 Welcome to the 2D Midpoint Program!!!

 What is the first end-point? (-3, 10)

 What is the second end-point? (-8, 10)

 Thank you!!! Calculating... Done.

 The midpoint of the line segment between (-3, 10) and (-8, 10)
 is (-5.5, 10).

Thank you for using the TMP!!

Have a good day
ee43254@ares:~$ ./midpoint.out

 Welcome to the 2D Midpoint Program!!!

 What is the first end-point? $3.4, 12.2*

 What is the second end-point? @13.4, 12.2(

 Thank you!!! Calculating... Done.

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

Thank you for using the TMP!!

Have a good day
ee43254@ares:~$ cat midpoint.tpq
TPQs
1. How many variables are needed here?

        Six variables are needed. Three each for taking in the point and
        calculating a midpoint x or y.
```

```
2. Would it make any sense to have whole-valued point coordinates?

        It would not be correct to assume that a point's coordinates would never
        have fractional parts. The user may input decimal or short data types but
        it would be better to have the ability to take both which double offers.
        This is done by dividing the points by 2.0 so it always returns as a doubl
        regardless if the user input a short or a double. This is also fixed by
        having the data types as doubles.

3. What is the formula to calculate a midpoint?

        The formula to calculate a midpoint would be midx = (x1+x2)/2.0 and midy =
        (y1+y2)/2.0. This would give the midpoint of x and y and thus your
        midpoint.

4. What happens if the input points are in different quadrants? In the same
quadrant? On an axis? At the origin?

        The way I have my program set up makes it so that regardless of the
        quadrant it will always correctly calculate the midpoint.

5. How many different ways can you pick a pair of points from the Cartesian
plane?

        It would be nine different ways to pick a pair of points. It would be four
        from each quadrant, four from each axes, and one from the origin.

More TPQs

1. To make the input style more natural, you'll need to make the user type in
the parentheses and the comma for typical Cartesian point notation. What [data]
type of variable(s) will this require?

        A char data type will be required to store the input.

2. How many variables will you need to make the input style more natural for
the user? At minimum? At most?

        At minimum, I would only need one char variable. At most, it would be six
        char variables.

3. How can the parentheses and comma be placed right next to the numbers like
that? How can cin determine where the number stops and the other stuff begins?

        By manipulating the cin line, one can force the input to take in a char an
        skip straight to the number.

4. What happens if the user doesn't type the parentheses or the comma? What
happens if they type some other symbols instead? Letters instead? Digits
instead?

        Since char variable stores any one character, it wouldn't matter if the
        user types a different symbol.

5. What if everything they type is non-numeric (i.e. '&*$#^')? Remember what
happened when you typed a symbol or letter at a numeric prompt before?

        It would skip over the second input and return a very small number.

        ee43254@ares:~$ exit
exit

Script done on 2023-09-25 15:30:40-05:00 [COMMAND_EXIT_CODE="0"]
```