CS1010E Practical Assessment #2 (Question)

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Local Extremum

Topic Coverage

- Assignment and expressions
- Control statements
- Functions and procedures

Problem Description

The **local maximum** of a function is a point `(x,y)` on the graph of the function whose `y` coordinate is larger than all other corresponding coordinates near `(x,y)`. Specifically, within an interval `[a,b]` that contains `x`, `(x,f(x))` is a local maximum if `f(x) > f(z)` for every $z \neq x$ in the interval `[a,b]`.

The example given below shows how we can find the local maximum $(x_{max}, p(x_{max}))$ of a polynomial `p(x)` within the range $[x_L, x_H]$ where `p(x)` is observed to be unimodal, i.e. `p(x)` is strictly increasing for values $x < x_{max}$ and `p(x)` is strictly decreasing for values of $x > x_{max}$.

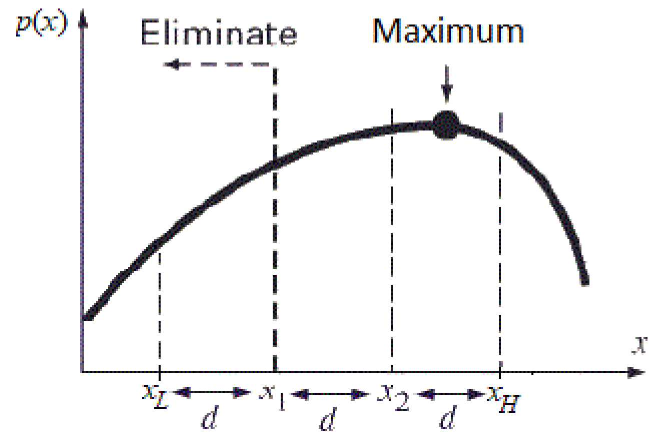


Figure (a)

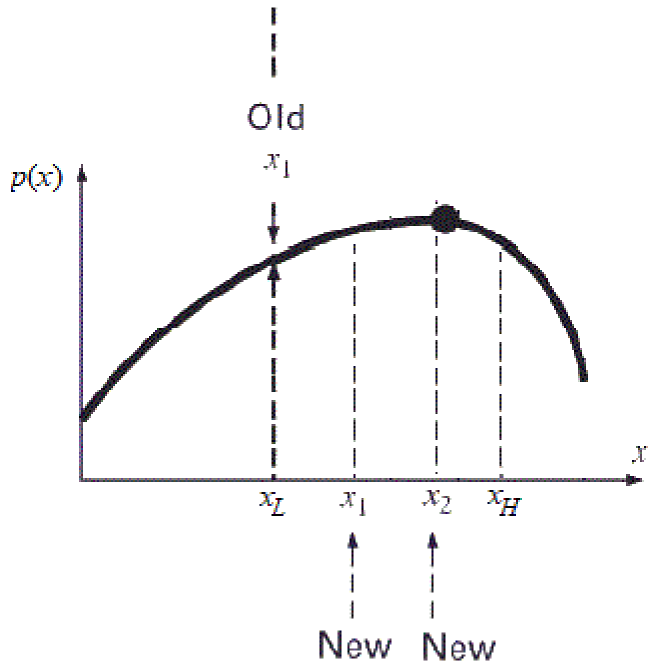


Figure (b)

As shown in Figure (a), we divide the interval into three equal sections by first finding two points x_1 and x_2 which lie at one-third intervals within x_L and x_H. Specifically,

$x_1 = x_L + d$

and

$x_2 = x_H - d$

We compare the values of x_1 and x_2 evaluated at the polynomial, i.e. $p(x_1)$ and $p(x_2)$. Notice that since $p(x_1) < p(x_2)$, we are assured that the maximum point will not lie within the leftmost section given by the interval $[x_L, x_1]$, and we can eliminate it.

In the next iteration in Figure (b), we let x_L be the old x_1 value from the previous iteration and determine the new x_1 and x_2 values at one-third intervals yet again. Comparing the values of $p(x_1)$ and $p(x_2)$ will again allow us to eliminate the leftmost section.

Hence, through repeated eliminations, the interval will eventually become so small that you will be able to approximate the location of the local maximum at x_{max} and it's corresponding maximum value $p(x_{max})$.

As another example of eliminating the rightmost section, Figure (c) below shows the case where the local maximum is located nearer to the left side of the interval. In this case, since $p(x_1) \geq p(x_2)$ (or in other words, $p(x_1) < p(x_2)$ is violated), we are assured that the maximum point will not lie within the rightmost section given by the interval $[x_2, x_H]$, and it can be eliminated.

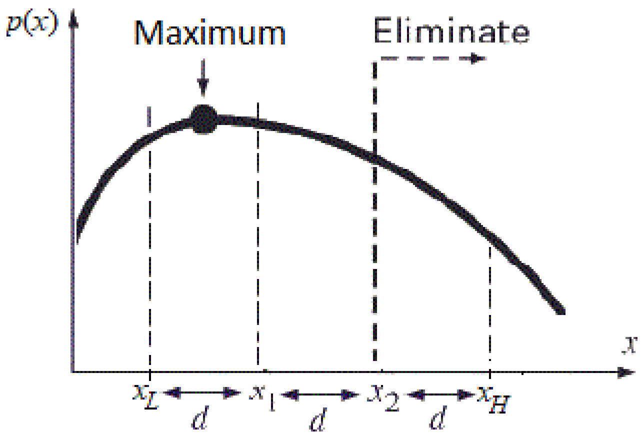
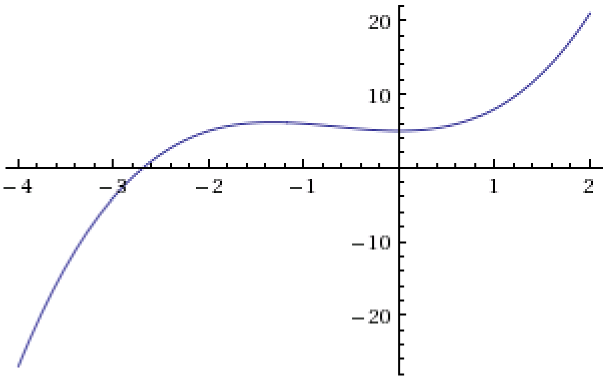


Figure (c)

Now, a more concrete example using the polynomial `p(x) = x`^{3} `+ 2x`^{2} `+ 5` to illustrate the process of finding the local maximum.



You are given that a local maximum lies within the interval `[-2,-1]`. In the first iteration, since x_L `= -2` and x_H `= -1`, we can compute x_1 `= -1.666667` and x_2 `= -1.333333`, as well as the corresponding values of $p(x_1)$ `= 5.925926` and $p(x_2)$ `= 6.185185`. Since $p(x_1) < p(x_2)$, we can eliminate the leftmost section of the interval.

In the next iteration, x_L `= -1.666667` and x_H `= -1`, we compute x_1 `= -1.444444` and x_2 `= -1.222222`, we well as the corresponding values of $p(x_1)$ `= 6.159122` and $p(x_2)$ `= 6.161866`. Since $p(x_1) < p(x_2)$, we can eliminate the leftmost section again.

In the third iteration, x_L `= -1.444444` and x_H `= -1`, we compute x_1 `= -1.296296` and x_2 `= -1.148148`, we well as the corresponding values of $p(x_1)$ `= 6.182493` and $p(x_2)$ `= 6.122949`. Since this time $p(x_1) \geq p(x_2)$, we eliminate the rightmost section. This results in $[x_L, x_H]$ `= [-1.444444, -1.148148]` for the next iteration.

Continuing this way, we obtain the following table of values for all iterations:

iteration	x_L	x_H	x_1	x_2	$p(x_1)$	$p(x_2)$
1	-2.000000	-1.000000	-1.666667	-1.333333	5.925926	6.185185
2	-1.666667	-1.000000	-1.444444	-1.222222	6.159122	6.161866
3	-1.444444	-1.000000	-1.296296	-1.148148	6.182493	6.122949
4	-1.444444	-1.148148	-1.345679	-1.246914	6.184878	6.170894
5	-1.444444	-1.246914	-1.378601	-1.312757	6.180994	6.184347
6	-1.378601	-1.246914	-1.334705	-1.290809	6.185181	6.181645
7	-1.378601	-1.290809	-1.349337	-1.320073	6.184669	6.184836
8	-1.349337	-1.290809	-1.329828	-1.310319	6.185161	6.184138
9	-1.349337	-1.310319	-1.336331	-1.323325	6.185167	6.184986
10	-1.349337	-1.323325	-1.340666	-1.331995	6.185077	6.185182
11	-1.340666	-1.323325	-1.334886	-1.329105	6.185180	6.185150
12	-1.340666	-1.329105	-1.336813	-1.332959	6.185161	6.185185

13	-1.336813	-1.329105	-1.334243	-1.331674	6.185184	6.185180
14	-1.336813	-1.331674	-1.335100	-1.333387	6.185179	6.185185
15	-1.335100	-1.331674	-1.333958	-1.332816	6.185184	6.185185
16	-1.333958	-1.331674	-1.333197	-1.332436	6.185185	6.185184
17	-1.333958	-1.332436	-1.333451	-1.332943	6.185185	6.185185
18	-1.333958	-1.332943	-1.333620	-1.333281	6.185185	6.185185
19	-1.333620	-1.332943	-1.333394	-1.333169	6.185185	6.185185
20	-1.333620	-1.333169	-1.333469	-1.333319	6.185185	6.185185
21	-1.333469	-1.333169	-1.333369	-1.333269	6.185185	6.185185
22	-1.333469	-1.333269	-1.333402	-1.333336	6.185185	6.185185
23	-1.333402	-1.333269	-1.333358	-1.333313	6.185185	6.185185
24	-1.333358	-1.333269				

Note that the algorithm terminates when $x_H - x_L < 0.0001$. The final approximate location of the local maximum x_{max} is given by the midway point between x_L and x_H. In the above example, the local maximum is `(-1.333313,6.185185)`.

In a similar way, if the given interval contains a **local minimum**, then we can iteratively find the local minimum point by eliminating the leftmost section of the interval if $p(x_1) > p(x_2)$. Otherwise, we eliminate the rightmost section.

Task

Write a program that asks the user to enter the integer coefficients (c_3, c_2, c_1, c_0) for a polynomial of degree 3:

$$c_3x^3 + c_2x^2 + c_1x + c_0$$

It then asks for the interval $[x_L, x_H]$, which are real numbers.

The last input to the program is either `1` to denote finding the local maximum, or `-1` to denote finding the local minimum within the given interval.

Take note of the following:

- Assume that the user enters a polynomial function of at least degree 2, i.e. a quadratic polynomial.
- Assume that the given interval contains exactly ONE local minimum or local maximum.
- Use the **double** data type for real numbers.
- Do not worry about the difference between `0.000000` and `-0.000000` when comparing output; these will be deemed the same in CodeCrunch.
- Suggested function declarations are provided; you may choose to use, modify or ignore them.
- You may use any C Math library function if deemed necessary.

This task is divided into several levels. Read through all the levels (from first to last, then from last to first) to see how the different levels are related. **You may start from any level.**

Level 1

Name your program `local1.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre interval $[x_L, x_H]$. Output the values of the coefficients and interval on separate lines.

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
1 2 0 5
xL = -2.000000; xH = -1.000000
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local1.out
```

To proceed to the next level (say level 2), copy your program by typing the Unix command

```
cp local1.c local2.c
```

Level 2

Name your program `local2.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre
interval $[x_L, x_H]$. Output the values of the coefficients.

Finally, output the values of the interval x_L and x_H, and the one-third interval values x_1 and x_2.

You may define the following function:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval $[x_L,x_H]$.

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
1 2 0 5
xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local2.out
```

To proceed to the next level (say level 3), copy your program by typing the Unix command

```
cp local2.c local3.c
```

Level 3

Name your program `local3.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre
interval $[x_L, x_H]$.

Output the values of the interval x_L and x_H, the one-third interval values x_1 and x_2, as well as $p(x_1)$ and $p(x_2)$.

You may define the following functions:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval $[x_L,x_H]$.

`double polynomial(double x, int c3, int c2, int c1, int c0);`
Evaluates and returns the value of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = 5.925926; p(x2) = 6.18518
```

```
$ ./a.out
0 -1 1 1
-1.0 1.0
xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = 0.555556; p(x2) = 1.222222
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local3.out
```

To proceed to the next level (say level 4), copy your program by typing the Unix command

```
cp local3.c local4.c
```

Level 4

Name your program `local4.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre
interval $[x_L, x_H]$.

Output the original values of the interval x_L and x_H, the one-third interval values x_1 and x_2, as well as $p(x_1)$ and $p(x_2)$.

Finally, perform one iteration of the search for the local maximum and eliminate either the leftmost or rightmost section.

Output the new values of x_L and x_H on a separate line.

You may define the following functions:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval $[x_L,x_H]$.

`double polynomial(double x, int c3, int c2, int c1, int c0);`
Evaluates and returns the value of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = 5.925926; p(x2) = 6.18518
xL = -1.666667; xH = -1.000000
```

```
$ ./a.out
1 2 0 5
-2.0 0
xL = -2.000000; xH = 0.000000; x1 = -1.333333; x2 = -0.666667; p(x1) = 6.185185; p(x2) = 5.59259
xL = -2.000000; xH = -0.666667
```

```
$ ./a.out
0 -1 1 1
-1.0 1.0
xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = 0.555556; p(x2) = 1.222222
xL = -0.333333; xH = 1.000000
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local4.out
```

To proceed to the next level (say level 5), copy your program by typing the Unix command

```
cp local4.c local5.c
```

Level 5

Name your program `local5.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre
interval [x_L, x_H].

Go through each iteration of the search for the local maximum. During each iteration, output the iteration number, the valu
one-third interval values x_1 and x_2, as well as $p(x_1)$ and $p(x_2)$. In addition, eliminate either the leftmost or rightmost sect
of x_L and x_H for the next iteration.

Terminate the iteration when the most recent values of $x_H - x_L < 0.0001$, and output these values on a separate line.

You may define the following functions:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval [x_L,x_H].

`double polynomial(double x, int c3, int c2, int c1, int c0);`
Evaluates and returns the value of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
iter = 1; xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = 5.925926; p(x2)
iter = 2; xL = -1.666667; xH = -1.000000; x1 = -1.444444; x2 = -1.222222; p(x1) = 6.159122; p(x2)
iter = 3; xL = -1.444444; xH = -1.000000; x1 = -1.296296; x2 = -1.148148; p(x1) = 6.182493; p(x2)
iter = 4; xL = -1.444444; xH = -1.148148; x1 = -1.345679; x2 = -1.246914; p(x1) = 6.184878; p(x2)
iter = 5; xL = -1.444444; xH = -1.246914; x1 = -1.378601; x2 = -1.312757; p(x1) = 6.180994; p(x2)
iter = 6; xL = -1.378601; xH = -1.246914; x1 = -1.334705; x2 = -1.290809; p(x1) = 6.185181; p(x2)
iter = 7; xL = -1.378601; xH = -1.290809; x1 = -1.349337; x2 = -1.320073; p(x1) = 6.184669; p(x2)
iter = 8; xL = -1.349337; xH = -1.290809; x1 = -1.329828; x2 = -1.310319; p(x1) = 6.185161; p(x2)
iter = 9; xL = -1.349337; xH = -1.310319; x1 = -1.336331; x2 = -1.323325; p(x1) = 6.185167; p(x2)
iter = 10; xL = -1.349337; xH = -1.323325; x1 = -1.340666; x2 = -1.331995; p(x1) = 6.185077; p(x2
iter = 11; xL = -1.340666; xH = -1.323325; x1 = -1.334886; x2 = -1.329105; p(x1) = 6.185180; p(x2
iter = 12; xL = -1.340666; xH = -1.329105; x1 = -1.336813; x2 = -1.332959; p(x1) = 6.185161; p(x2
iter = 13; xL = -1.336813; xH = -1.329105; x1 = -1.334243; x2 = -1.331674; p(x1) = 6.185184; p(x2
iter = 14; xL = -1.336813; xH = -1.331674; x1 = -1.335100; x2 = -1.333387; p(x1) = 6.185179; p(x2
iter = 15; xL = -1.335100; xH = -1.331674; x1 = -1.333958; x2 = -1.332816; p(x1) = 6.185184; p(x2
iter = 16; xL = -1.333958; xH = -1.331674; x1 = -1.333197; x2 = -1.332436; p(x1) = 6.185185; p(x2
iter = 17; xL = -1.333958; xH = -1.332436; x1 = -1.333451; x2 = -1.332943; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.333958; xH = -1.332943; x1 = -1.333620; x2 = -1.333281; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333620; xH = -1.332943; x1 = -1.333394; x2 = -1.333169; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333620; xH = -1.333169; x1 = -1.333469; x2 = -1.333319; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333469; xH = -1.333169; x1 = -1.333369; x2 = -1.333269; p(x1) = 6.185185; p(x2
iter = 22; xL = -1.333469; xH = -1.333269; x1 = -1.333402; x2 = -1.333336; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333402; xH = -1.333269; x1 = -1.333358; x2 = -1.333313; p(x1) = 6.185185; p(x2
xL = -1.333358; xH = -1.333269
```

```
$ ./a.out
1 2 0 5
-2.0 0
iter = 1; xL = -2.000000; xH = 0.000000; x1 = -1.333333; x2 = -0.666667; p(x1) = 6.185185; p(x2)
iter = 2; xL = -2.000000; xH = -0.666667; x1 = -1.555556; x2 = -1.111111; p(x1) = 6.075446; p(x2)
iter = 3; xL = -1.555556; xH = -0.666667; x1 = -1.259259; x2 = -0.962963; p(x1) = 6.174618; p(x2)
iter = 4; xL = -1.555556; xH = -0.962963; x1 = -1.358025; x2 = -1.160494; p(x1) = 6.183951; p(x2)
iter = 5; xL = -1.555556; xH = -1.160494; x1 = -1.423868; x2 = -1.292181; p(x1) = 6.168050; p(x2)
iter = 6; xL = -1.423868; xH = -1.160494; x1 = -1.336077; x2 = -1.248285; p(x1) = 6.185170; p(x2)
iter = 7; xL = -1.423868; xH = -1.248285; x1 = -1.365341; x2 = -1.306813; p(x1) = 6.183103; p(x2)
iter = 8; xL = -1.365341; xH = -1.248285; x1 = -1.326322; x2 = -1.287304; p(x1) = 6.185087; p(x2)
iter = 9; xL = -1.365341; xH = -1.287304; x1 = -1.339328; x2 = -1.313316; p(x1) = 6.185113; p(x2)
iter = 10; xL = -1.365341; xH = -1.313316; x1 = -1.347999; x2 = -1.330658; p(x1) = 6.184752; p(x2
iter = 11; xL = -1.347999; xH = -1.313316; x1 = -1.336438; x2 = -1.324877; p(x1) = 6.185166; p(x2
iter = 12; xL = -1.347999; xH = -1.324877; x1 = -1.340292; x2 = -1.332584; p(x1) = 6.185088; p(x2
iter = 13; xL = -1.340292; xH = -1.324877; x1 = -1.335154; x2 = -1.330015; p(x1) = 6.185179; p(x2
iter = 14; xL = -1.340292; xH = -1.330015; x1 = -1.336866; x2 = -1.333441; p(x1) = 6.185160; p(x2
iter = 15; xL = -1.336866; xH = -1.330015; x1 = -1.334583; x2 = -1.332299; p(x1) = 6.185182; p(x2
iter = 16; xL = -1.334583; xH = -1.330015; x1 = -1.333060; x2 = -1.331538; p(x1) = 6.185185; p(x2
iter = 17; xL = -1.334583; xH = -1.331538; x1 = -1.333568; x2 = -1.332553; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.334583; xH = -1.332553; x1 = -1.333906; x2 = -1.333229; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333906; xH = -1.332553; x1 = -1.333455; x2 = -1.333004; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333906; xH = -1.333004; x1 = -1.333605; x2 = -1.333305; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333605; xH = -1.333004; x1 = -1.333405; x2 = -1.333204; p(x1) = 6.185185; p(x2
```

```
iter = 22; xL = -1.333605; xH = -1.333204; x1 = -1.333472; x2 = -1.333338; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333472; xH = -1.333204; x1 = -1.333382; x2 = -1.333293; p(x1) = 6.185185; p(x2
iter = 24; xL = -1.333382; xH = -1.333204; x1 = -1.333323; x2 = -1.333264; p(x1) = 6.185185; p(x2
iter = 25; xL = -1.333382; xH = -1.333264; x1 = -1.333343; x2 = -1.333303; p(x1) = 6.185185; p(x2
xL = -1.333382; xH = -1.333303
```

```
$ ./a.out
0 -1 1 1
-1.0 1.0
iter = 1; xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = 0.555556; p(x2) =
iter = 2; xL = -0.333333; xH = 1.000000; x1 = 0.111111; x2 = 0.555556; p(x1) = 1.098765; p(x2) =
iter = 3; xL = 0.111111; xH = 1.000000; x1 = 0.407407; x2 = 0.703704; p(x1) = 1.241427; p(x2) = 1
iter = 4; xL = 0.111111; xH = 0.703704; x1 = 0.308642; x2 = 0.506173; p(x1) = 1.213382; p(x2) = 1
iter = 5; xL = 0.308642; xH = 0.703704; x1 = 0.440329; x2 = 0.572016; p(x1) = 1.246439; p(x2) = 1
iter = 6; xL = 0.308642; xH = 0.572016; x1 = 0.396433; x2 = 0.484225; p(x1) = 1.239274; p(x2) = 1
iter = 7; xL = 0.396433; xH = 0.572016; x1 = 0.454961; x2 = 0.513489; p(x1) = 1.247972; p(x2) = 1
iter = 8; xL = 0.454961; xH = 0.572016; x1 = 0.493980; x2 = 0.532998; p(x1) = 1.249964; p(x2) = 1
iter = 9; xL = 0.454961; xH = 0.532998; x1 = 0.480973; x2 = 0.506986; p(x1) = 1.249638; p(x2) = 1
iter = 10; xL = 0.480973; xH = 0.532998; x1 = 0.498315; x2 = 0.515656; p(x1) = 1.249997; p(x2) =
iter = 11; xL = 0.480973; xH = 0.515656; x1 = 0.492534; x2 = 0.504095; p(x1) = 1.249944; p(x2) =
iter = 12; xL = 0.492534; xH = 0.515656; x1 = 0.500242; x2 = 0.507949; p(x1) = 1.250000; p(x2) =
iter = 13; xL = 0.492534; xH = 0.507949; x1 = 0.497673; x2 = 0.502811; p(x1) = 1.249995; p(x2) =
iter = 14; xL = 0.492534; xH = 0.502811; x1 = 0.495960; x2 = 0.499385; p(x1) = 1.249984; p(x2) =
iter = 15; xL = 0.495960; xH = 0.502811; x1 = 0.498244; x2 = 0.500527; p(x1) = 1.249997; p(x2) =
iter = 16; xL = 0.498244; xH = 0.502811; x1 = 0.499766; x2 = 0.501288; p(x1) = 1.250000; p(x2) =
iter = 17; xL = 0.498244; xH = 0.501288; x1 = 0.499259; x2 = 0.500274; p(x1) = 1.249999; p(x2) =
iter = 18; xL = 0.499259; xH = 0.501288; x1 = 0.499935; x2 = 0.500612; p(x1) = 1.250000; p(x2) =
iter = 19; xL = 0.499259; xH = 0.500612; x1 = 0.499710; x2 = 0.500161; p(x1) = 1.250000; p(x2) =
iter = 20; xL = 0.499710; xH = 0.500612; x1 = 0.500010; x2 = 0.500311; p(x1) = 1.250000; p(x2) =
iter = 21; xL = 0.499710; xH = 0.500311; x1 = 0.499910; x2 = 0.500111; p(x1) = 1.250000; p(x2) =
iter = 22; xL = 0.499710; xH = 0.500111; x1 = 0.499843; x2 = 0.499977; p(x1) = 1.250000; p(x2) =
iter = 23; xL = 0.499843; xH = 0.500111; x1 = 0.499932; x2 = 0.500022; p(x1) = 1.250000; p(x2) =
iter = 24; xL = 0.499932; xH = 0.500111; x1 = 0.499992; x2 = 0.500051; p(x1) = 1.250000; p(x2) =
iter = 25; xL = 0.499932; xH = 0.500051; x1 = 0.499972; x2 = 0.500012; p(x1) = 1.250000; p(x2) =
xL = 0.499972; xH = 0.500051
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local5.out
```

To proceed to the next level (say level 6), copy your program by typing the Unix command

```
cp local5.c local6.c
```

Level 6

Name your program `local6.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre
interval $[x_L, x_H]$.

Go through each iteration of the search for the local maximum. During each iteration, output the iteration number, the valu
one-third interval values x_1 and x_2, as well as $p(x_1)$ and $p(x_2)$. In addition, eliminate either the leftmost or rightmost sect
of x_L and x_H for the next iteration.

Terminate the iteration when $x_H - x_L < 0.0001$. Output the middle point between x_L and x_H as the approximate location
output the corresponding $p(x_{max})$ on a separate line.

You may define the following functions:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval $[x_L,x_H]$.

`double polynomial(double x, int c3, int c2, int c1, int c0);`
Evaluates and returns the value of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$

`double search(int c3, int c2, int c1, int c0, double xL, double xH);`

Performs the search for local maximum of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$ within the interval $[x_L,x_H]$. Th
x_{max} is returned.

Tip: Although this level is relatively straightforward, you are advised to give some thought to defining the `search` function. function look more elegant, but it might provide you a hint as to how best to proceed to the next level.

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
iter = 1; xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = 5.925926; p(x2)
iter = 2; xL = -1.666667; xH = -1.000000; x1 = -1.444444; x2 = -1.222222; p(x1) = 6.159122; p(x2)
iter = 3; xL = -1.444444; xH = -1.000000; x1 = -1.296296; x2 = -1.148148; p(x1) = 6.182493; p(x2)
iter = 4; xL = -1.444444; xH = -1.148148; x1 = -1.345679; x2 = -1.246914; p(x1) = 6.184878; p(x2)
iter = 5; xL = -1.444444; xH = -1.246914; x1 = -1.378601; x2 = -1.312757; p(x1) = 6.180994; p(x2)
iter = 6; xL = -1.378601; xH = -1.246914; x1 = -1.334705; x2 = -1.290809; p(x1) = 6.185181; p(x2)
iter = 7; xL = -1.378601; xH = -1.290809; x1 = -1.349337; x2 = -1.320073; p(x1) = 6.184669; p(x2)
iter = 8; xL = -1.349337; xH = -1.290809; x1 = -1.329828; x2 = -1.310319; p(x1) = 6.185161; p(x2)
iter = 9; xL = -1.349337; xH = -1.310319; x1 = -1.336331; x2 = -1.323325; p(x1) = 6.185167; p(x2)
iter = 10; xL = -1.349337; xH = -1.323325; x1 = -1.340666; x2 = -1.331995; p(x1) = 6.185077; p(x2
iter = 11; xL = -1.340666; xH = -1.323325; x1 = -1.334886; x2 = -1.329105; p(x1) = 6.185180; p(x2
iter = 12; xL = -1.340666; xH = -1.329105; x1 = -1.336813; x2 = -1.332959; p(x1) = 6.185161; p(x2
iter = 13; xL = -1.336813; xH = -1.329105; x1 = -1.334243; x2 = -1.331674; p(x1) = 6.185184; p(x2
iter = 14; xL = -1.336813; xH = -1.331674; x1 = -1.335100; x2 = -1.333387; p(x1) = 6.185179; p(x2
iter = 15; xL = -1.335100; xH = -1.331674; x1 = -1.333958; x2 = -1.332816; p(x1) = 6.185184; p(x2
iter = 16; xL = -1.333958; xH = -1.331674; x1 = -1.333197; x2 = -1.332436; p(x1) = 6.185185; p(x2
iter = 17; xL = -1.333958; xH = -1.332436; x1 = -1.333451; x2 = -1.332943; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.333958; xH = -1.332943; x1 = -1.333620; x2 = -1.333281; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333620; xH = -1.332943; x1 = -1.333394; x2 = -1.333169; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333620; xH = -1.333169; x1 = -1.333469; x2 = -1.333319; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333469; xH = -1.333169; x1 = -1.333369; x2 = -1.333269; p(x1) = 6.185185; p(x2
iter = 22; xL = -1.333469; xH = -1.333269; x1 = -1.333402; x2 = -1.333336; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333402; xH = -1.333269; x1 = -1.333358; x2 = -1.333313; p(x1) = 6.185185; p(x2
Local maximum point is (-1.333313, 6.185185)
```

```
$ ./a.out
1 2 0 5
-2.0 0
iter = 1; xL = -2.000000; xH = 0.000000; x1 = -1.333333; x2 = -0.666667; p(x1) = 6.185185; p(x2)
iter = 2; xL = -2.000000; xH = -0.666667; x1 = -1.555556; x2 = -1.111111; p(x1) = 6.075446; p(x2)
iter = 3; xL = -1.555556; xH = -0.666667; x1 = -1.259259; x2 = -0.962963; p(x1) = 6.174618; p(x2)
iter = 4; xL = -1.555556; xH = -0.962963; x1 = -1.358025; x2 = -1.160494; p(x1) = 6.183951; p(x2)
iter = 5; xL = -1.555556; xH = -1.160494; x1 = -1.423868; x2 = -1.292181; p(x1) = 6.168050; p(x2)
iter = 6; xL = -1.423868; xH = -1.160494; x1 = -1.336077; x2 = -1.248285; p(x1) = 6.185170; p(x2)
iter = 7; xL = -1.423868; xH = -1.248285; x1 = -1.365341; x2 = -1.306813; p(x1) = 6.183103; p(x2)
iter = 8; xL = -1.365341; xH = -1.248285; x1 = -1.326322; x2 = -1.287304; p(x1) = 6.185087; p(x2)
iter = 9; xL = -1.365341; xH = -1.287304; x1 = -1.339328; x2 = -1.313316; p(x1) = 6.185113; p(x2)
iter = 10; xL = -1.365341; xH = -1.313316; x1 = -1.347999; x2 = -1.330658; p(x1) = 6.184752; p(x2
iter = 11; xL = -1.347999; xH = -1.313316; x1 = -1.336438; x2 = -1.324877; p(x1) = 6.185166; p(x2
iter = 12; xL = -1.347999; xH = -1.324877; x1 = -1.340292; x2 = -1.332584; p(x1) = 6.185088; p(x2
iter = 13; xL = -1.340292; xH = -1.324877; x1 = -1.335154; x2 = -1.330015; p(x1) = 6.185179; p(x2
iter = 14; xL = -1.340292; xH = -1.330015; x1 = -1.336866; x2 = -1.333441; p(x1) = 6.185160; p(x2
iter = 15; xL = -1.336866; xH = -1.330015; x1 = -1.334583; x2 = -1.332299; p(x1) = 6.185182; p(x2
iter = 16; xL = -1.334583; xH = -1.330015; x1 = -1.333060; x2 = -1.331538; p(x1) = 6.185185; p(x2
iter = 17; xL = -1.334583; xH = -1.331538; x1 = -1.333568; x2 = -1.332553; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.334583; xH = -1.332553; x1 = -1.333906; x2 = -1.333229; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333906; xH = -1.332553; x1 = -1.333455; x2 = -1.333004; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333906; xH = -1.333004; x1 = -1.333605; x2 = -1.333305; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333605; xH = -1.333004; x1 = -1.333405; x2 = -1.333204; p(x1) = 6.185185; p(x2
iter = 22; xL = -1.333605; xH = -1.333204; x1 = -1.333472; x2 = -1.333338; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333472; xH = -1.333204; x1 = -1.333382; x2 = -1.333293; p(x1) = 6.185185; p(x2
iter = 24; xL = -1.333382; xH = -1.333204; x1 = -1.333323; x2 = -1.333264; p(x1) = 6.185185; p(x2
iter = 25; xL = -1.333382; xH = -1.333264; x1 = -1.333343; x2 = -1.333303; p(x1) = 6.185185; p(x2
Local maximum point is (-1.333343, 6.185185)
```

```
$ ./a.out
0 -1 1 1
-1.0 1.0
iter = 1; xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = 0.555556; p(x2) =
iter = 2; xL = -0.333333; xH = 1.000000; x1 = 0.111111; x2 = 0.555556; p(x1) = 1.098765; p(x2) =
iter = 3; xL = 0.111111; xH = 1.000000; x1 = 0.407407; x2 = 0.703704; p(x1) = 1.241427; p(x2) = 1
iter = 4; xL = 0.111111; xH = 0.703704; x1 = 0.308642; x2 = 0.506173; p(x1) = 1.213382; p(x2) = 1
iter = 5; xL = 0.308642; xH = 0.703704; x1 = 0.440329; x2 = 0.572016; p(x1) = 1.246439; p(x2) = 1
```

```
iter = 6; xL = 0.308642; xH = 0.572016; x1 = 0.396433; x2 = 0.484225; p(x1) = 1.239274; p(x2) = 1
iter = 7; xL = 0.396433; xH = 0.572016; x1 = 0.454961; x2 = 0.513489; p(x1) = 1.247972; p(x2) = 1
iter = 8; xL = 0.454961; xH = 0.572016; x1 = 0.493980; x2 = 0.532998; p(x1) = 1.249964; p(x2) = 1
iter = 9; xL = 0.454961; xH = 0.532998; x1 = 0.480973; x2 = 0.506986; p(x1) = 1.249638; p(x2) = 1
iter = 10; xL = 0.480973; xH = 0.532998; x1 = 0.498315; x2 = 0.515656; p(x1) = 1.249997; p(x2) =
iter = 11; xL = 0.480973; xH = 0.515656; x1 = 0.492534; x2 = 0.504095; p(x1) = 1.249944; p(x2) =
iter = 12; xL = 0.492534; xH = 0.515656; x1 = 0.500242; x2 = 0.507949; p(x1) = 1.250000; p(x2) =
iter = 13; xL = 0.492534; xH = 0.507949; x1 = 0.497673; x2 = 0.502811; p(x1) = 1.249995; p(x2) =
iter = 14; xL = 0.492534; xH = 0.502811; x1 = 0.495960; x2 = 0.499385; p(x1) = 1.249984; p(x2) =
iter = 15; xL = 0.495960; xH = 0.502811; x1 = 0.498244; x2 = 0.500527; p(x1) = 1.249997; p(x2) =
iter = 16; xL = 0.498244; xH = 0.502811; x1 = 0.499766; x2 = 0.501288; p(x1) = 1.250000; p(x2) =
iter = 17; xL = 0.498244; xH = 0.501288; x1 = 0.499259; x2 = 0.500274; p(x1) = 1.249999; p(x2) =
iter = 18; xL = 0.499259; xH = 0.501288; x1 = 0.499935; x2 = 0.500612; p(x1) = 1.250000; p(x2) =
iter = 19; xL = 0.499259; xH = 0.500612; x1 = 0.499710; x2 = 0.500161; p(x1) = 1.250000; p(x2) =
iter = 20; xL = 0.499710; xH = 0.500612; x1 = 0.500010; x2 = 0.500311; p(x1) = 1.250000; p(x2) =
iter = 21; xL = 0.499710; xH = 0.500311; x1 = 0.499910; x2 = 0.500111; p(x1) = 1.250000; p(x2) =
iter = 22; xL = 0.499710; xH = 0.500111; x1 = 0.499843; x2 = 0.499977; p(x1) = 1.250000; p(x2) =
iter = 23; xL = 0.499843; xH = 0.500111; x1 = 0.499932; x2 = 0.500022; p(x1) = 1.250000; p(x2) =
iter = 24; xL = 0.499932; xH = 0.500111; x1 = 0.499992; x2 = 0.500051; p(x1) = 1.250000; p(x2) =
iter = 25; xL = 0.499932; xH = 0.500051; x1 = 0.499972; x2 = 0.500012; p(x1) = 1.250000; p(x2) =
Local maximum point is (0.500012, 1.250000)
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local6.out
```

To proceed to the next level (say level 7), copy your program by typing the Unix command

```
cp local6.c local7.c
```

Level 7

Name your program `local7.c`

Write a program that reads in four integer coefficients (c_3, c_2, c_1, c_0) followed by two floating-point values x_L and x_H repre interval [x_L, x_H]. The last input read is either a `1` for finding the local maximum, or `-1` for finding the local minimum.

Go through each iteration of the search for the local extremum (i.e. maximum or minimum). During each iteration, output t the interval x_L and x_H, the one-third interval values x_1 and x_2, as well as $p(x_1)$ and $p(x_2)$. In addition, eliminate either the compute the new values of x_L and x_H for the next iteration.

Terminate the iteration when $x_H - x_L < 0.0001$. Output the middle point between x_L and x_H as the approximate location output the corresponding $p(x_{ext})$ on a separate line. Note that you have to specify the type of local extremum (either ma

You may define the following functions:

`void findThirds(double xL, double xH, double *x1, double *x2);`
Compute the values of x_1 and x_2 given the interval [x_L,x_H].

`double polynomial(double x, int c3, int c2, int c1, int c0);`
Evaluates and returns the value of the polynomial $c_3x^3 + c_2x^2 + c_1x + c_0$

`double search(int c3, int c2, int c1, int c0, double xL, double xH, int maxmin);`
Performs the search for local maximum (`maxmin=1`) or minimum (`maxmin=-1`) of the polynomial $c_3x^3 + c_2x^2 + c$ [x_L,x_H]. The location of the local extremum x_{ext} is returned.

Tip: Compare the `search` function declaration with the one given in the preceding level.

You need not worry about the case where the user requests a maximum point to be found but the interval actually contain

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a

```
$ ./a.out
1 2 0 5
-2.0 -1.0
```

```
1
iter = 1; xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = 5.925926; p(x2)
iter = 2; xL = -1.666667; xH = -1.000000; x1 = -1.444444; x2 = -1.222222; p(x1) = 6.159122; p(x2)
iter = 3; xL = -1.444444; xH = -1.000000; x1 = -1.296296; x2 = -1.148148; p(x1) = 6.182493; p(x2)
iter = 4; xL = -1.444444; xH = -1.148148; x1 = -1.345679; x2 = -1.246914; p(x1) = 6.184878; p(x2)
iter = 5; xL = -1.444444; xH = -1.246914; x1 = -1.378601; x2 = -1.312757; p(x1) = 6.180994; p(x2)
iter = 6; xL = -1.378601; xH = -1.246914; x1 = -1.334705; x2 = -1.290809; p(x1) = 6.185181; p(x2)
iter = 7; xL = -1.378601; xH = -1.290809; x1 = -1.349337; x2 = -1.320073; p(x1) = 6.184669; p(x2)
iter = 8; xL = -1.349337; xH = -1.290809; x1 = -1.329828; x2 = -1.310319; p(x1) = 6.185161; p(x2)
iter = 9; xL = -1.349337; xH = -1.310319; x1 = -1.336331; x2 = -1.323325; p(x1) = 6.185167; p(x2)
iter = 10; xL = -1.349337; xH = -1.323325; x1 = -1.340666; x2 = -1.331995; p(x1) = 6.185077; p(x2
iter = 11; xL = -1.340666; xH = -1.323325; x1 = -1.334886; x2 = -1.329105; p(x1) = 6.185180; p(x2
iter = 12; xL = -1.340666; xH = -1.329105; x1 = -1.336813; x2 = -1.332959; p(x1) = 6.185161; p(x2
iter = 13; xL = -1.336813; xH = -1.329105; x1 = -1.334243; x2 = -1.331674; p(x1) = 6.185184; p(x2
iter = 14; xL = -1.336813; xH = -1.331674; x1 = -1.335100; x2 = -1.333387; p(x1) = 6.185179; p(x2
iter = 15; xL = -1.335100; xH = -1.331674; x1 = -1.333958; x2 = -1.332816; p(x1) = 6.185184; p(x2
iter = 16; xL = -1.333958; xH = -1.331674; x1 = -1.333197; x2 = -1.332436; p(x1) = 6.185185; p(x2
iter = 17; xL = -1.333958; xH = -1.332436; x1 = -1.333451; x2 = -1.332943; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.333958; xH = -1.332943; x1 = -1.333620; x2 = -1.333281; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333620; xH = -1.332943; x1 = -1.333394; x2 = -1.333169; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333620; xH = -1.333169; x1 = -1.333469; x2 = -1.333319; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333469; xH = -1.333169; x1 = -1.333369; x2 = -1.333269; p(x1) = 6.185185; p(x2
iter = 22; xL = -1.333469; xH = -1.333269; x1 = -1.333402; x2 = -1.333336; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333402; xH = -1.333269; x1 = -1.333358; x2 = -1.333313; p(x1) = 6.185185; p(x2
Local maximum point is (-1.333313, 6.185185)
```

```
$ ./a.out
-1 -2 0 -5
-2.0 -1.0
-1
iter = 1; xL = -2.000000; xH = -1.000000; x1 = -1.666667; x2 = -1.333333; p(x1) = -5.925926; p(x2
iter = 2; xL = -1.666667; xH = -1.000000; x1 = -1.444444; x2 = -1.222222; p(x1) = -6.159122; p(x2
iter = 3; xL = -1.444444; xH = -1.000000; x1 = -1.296296; x2 = -1.148148; p(x1) = -6.182493; p(x2
iter = 4; xL = -1.444444; xH = -1.148148; x1 = -1.345679; x2 = -1.246914; p(x1) = -6.184878; p(x2
iter = 5; xL = -1.444444; xH = -1.246914; x1 = -1.378601; x2 = -1.312757; p(x1) = -6.180994; p(x2
iter = 6; xL = -1.378601; xH = -1.246914; x1 = -1.334705; x2 = -1.290809; p(x1) = -6.185181; p(x2
iter = 7; xL = -1.378601; xH = -1.290809; x1 = -1.349337; x2 = -1.320073; p(x1) = -6.184669; p(x2
iter = 8; xL = -1.349337; xH = -1.290809; x1 = -1.329828; x2 = -1.310319; p(x1) = -6.185161; p(x2
iter = 9; xL = -1.349337; xH = -1.310319; x1 = -1.336331; x2 = -1.323325; p(x1) = -6.185167; p(x2
iter = 10; xL = -1.349337; xH = -1.323325; x1 = -1.340666; x2 = -1.331995; p(x1) = -6.185077; p(x
iter = 11; xL = -1.340666; xH = -1.323325; x1 = -1.334886; x2 = -1.329105; p(x1) = -6.185180; p(x
iter = 12; xL = -1.340666; xH = -1.329105; x1 = -1.336813; x2 = -1.332959; p(x1) = -6.185161; p(x
iter = 13; xL = -1.336813; xH = -1.329105; x1 = -1.334243; x2 = -1.331674; p(x1) = -6.185184; p(x
iter = 14; xL = -1.336813; xH = -1.331674; x1 = -1.335100; x2 = -1.333387; p(x1) = -6.185179; p(x
iter = 15; xL = -1.335100; xH = -1.331674; x1 = -1.333958; x2 = -1.332816; p(x1) = -6.185184; p(x
iter = 16; xL = -1.333958; xH = -1.331674; x1 = -1.333197; x2 = -1.332436; p(x1) = -6.185185; p(x
iter = 17; xL = -1.333958; xH = -1.332436; x1 = -1.333451; x2 = -1.332943; p(x1) = -6.185185; p(x
iter = 18; xL = -1.333958; xH = -1.332943; x1 = -1.333620; x2 = -1.333281; p(x1) = -6.185185; p(x
iter = 19; xL = -1.333620; xH = -1.332943; x1 = -1.333394; x2 = -1.333169; p(x1) = -6.185185; p(x
iter = 20; xL = -1.333620; xH = -1.333169; x1 = -1.333469; x2 = -1.333319; p(x1) = -6.185185; p(x
iter = 21; xL = -1.333469; xH = -1.333169; x1 = -1.333369; x2 = -1.333269; p(x1) = -6.185185; p(x
iter = 22; xL = -1.333469; xH = -1.333269; x1 = -1.333402; x2 = -1.333336; p(x1) = -6.185185; p(x
iter = 23; xL = -1.333402; xH = -1.333269; x1 = -1.333358; x2 = -1.333313; p(x1) = -6.185185; p(x
Local minimum point is (-1.333313, -6.185185)
```

```
$ ./a.out
1 2 0 5
-2.0 0
1
iter = 1; xL = -2.000000; xH = 0.000000; x1 = -1.333333; x2 = -0.666667; p(x1) = 6.185185; p(x2)
iter = 2; xL = -2.000000; xH = -0.666667; x1 = -1.555556; x2 = -1.111111; p(x1) = 6.075446; p(x2)
iter = 3; xL = -1.555556; xH = -0.666667; x1 = -1.259259; x2 = -0.962963; p(x1) = 6.174618; p(x2)
iter = 4; xL = -1.555556; xH = -0.962963; x1 = -1.358025; x2 = -1.160494; p(x1) = 6.183951; p(x2)
iter = 5; xL = -1.555556; xH = -1.160494; x1 = -1.423868; x2 = -1.292181; p(x1) = 6.168050; p(x2)
iter = 6; xL = -1.423868; xH = -1.160494; x1 = -1.336077; x2 = -1.248285; p(x1) = 6.185170; p(x2)
iter = 7; xL = -1.423868; xH = -1.248285; x1 = -1.365341; x2 = -1.306813; p(x1) = 6.183103; p(x2)
iter = 8; xL = -1.365341; xH = -1.248285; x1 = -1.326322; x2 = -1.287304; p(x1) = 6.185087; p(x2)
iter = 9; xL = -1.365341; xH = -1.287304; x1 = -1.339328; x2 = -1.313316; p(x1) = 6.185113; p(x2)
iter = 10; xL = -1.365341; xH = -1.313316; x1 = -1.347999; x2 = -1.330658; p(x1) = 6.184752; p(x2
iter = 11; xL = -1.347999; xH = -1.313316; x1 = -1.336438; x2 = -1.324877; p(x1) = 6.185166; p(x2
iter = 12; xL = -1.347999; xH = -1.324877; x1 = -1.340292; x2 = -1.332584; p(x1) = 6.185088; p(x2
iter = 13; xL = -1.340292; xH = -1.324877; x1 = -1.335154; x2 = -1.330015; p(x1) = 6.185179; p(x2
iter = 14; xL = -1.340292; xH = -1.330015; x1 = -1.336866; x2 = -1.333441; p(x1) = 6.185160; p(x2
iter = 15; xL = -1.336866; xH = -1.330015; x1 = -1.334583; x2 = -1.332299; p(x1) = 6.185182; p(x2
iter = 16; xL = -1.334583; xH = -1.330015; x1 = -1.333060; x2 = -1.331538; p(x1) = 6.185185; p(x2
```

```
iter = 17; xL = -1.334583; xH = -1.331538; x1 = -1.333568; x2 = -1.332553; p(x1) = 6.185185; p(x2
iter = 18; xL = -1.334583; xH = -1.332553; x1 = -1.333906; x2 = -1.333229; p(x1) = 6.185185; p(x2
iter = 19; xL = -1.333906; xH = -1.332553; x1 = -1.333455; x2 = -1.333004; p(x1) = 6.185185; p(x2
iter = 20; xL = -1.333906; xH = -1.333004; x1 = -1.333605; x2 = -1.333305; p(x1) = 6.185185; p(x2
iter = 21; xL = -1.333605; xH = -1.333004; x1 = -1.333405; x2 = -1.333204; p(x1) = 6.185185; p(x2
iter = 22; xL = -1.333605; xH = -1.333204; x1 = -1.333472; x2 = -1.333338; p(x1) = 6.185185; p(x2
iter = 23; xL = -1.333472; xH = -1.333204; x1 = -1.333382; x2 = -1.333293; p(x1) = 6.185185; p(x2
iter = 24; xL = -1.333382; xH = -1.333204; x1 = -1.333323; x2 = -1.333264; p(x1) = 6.185185; p(x2
iter = 25; xL = -1.333382; xH = -1.333264; x1 = -1.333343; x2 = -1.333303; p(x1) = 6.185185; p(x2
Local maximum point is (-1.333343, 6.185185)
```

```
$ ./a.out
1 2 0 5
-1.0 2.0
-1
iter = 1; xL = -1.000000; xH = 2.000000; x1 = 0.000000; x2 = 1.000000; p(x1) = 5.000000; p(x2) =
iter = 2; xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = 5.185185; p(x2) =
iter = 3; xL = -1.000000; xH = 0.333333; x1 = -0.555556; x2 = -0.111111; p(x1) = 5.445816; p(x2)
iter = 4; xL = -0.555556; xH = 0.333333; x1 = -0.259259; x2 = 0.037037; p(x1) = 5.117005; p(x2) =
iter = 5; xL = -0.259259; xH = 0.333333; x1 = -0.061728; x2 = 0.135802; p(x1) = 5.007386; p(x2) =
iter = 6; xL = -0.259259; xH = 0.135802; x1 = -0.127572; x2 = 0.004115; p(x1) = 5.030473; p(x2) =
iter = 7; xL = -0.127572; xH = 0.135802; x1 = -0.039781; x2 = 0.048011; p(x1) = 5.003102; p(x2) =
iter = 8; xL = -0.127572; xH = 0.048011; x1 = -0.069044; x2 = -0.010517; p(x1) = 5.009205; p(x2)
iter = 9; xL = -0.069044; xH = 0.048011; x1 = -0.030026; x2 = 0.008993; p(x1) = 5.001776; p(x2) =
iter = 10; xL = -0.030026; xH = 0.048011; x1 = -0.004014; x2 = 0.021999; p(x1) = 5.000032; p(x2)
iter = 11; xL = -0.030026; xH = 0.021999; x1 = -0.012684; x2 = 0.004657; p(x1) = 5.000320; p(x2)
iter = 12; xL = -0.012684; xH = 0.021999; x1 = -0.001123; x2 = 0.010438; p(x1) = 5.000003; p(x2)
iter = 13; xL = -0.012684; xH = 0.010438; x1 = -0.004977; x2 = 0.002730; p(x1) = 5.000049; p(x2)
iter = 14; xL = -0.004977; xH = 0.010438; x1 = 0.000161; x2 = 0.005299; p(x1) = 5.000000; p(x2) =
iter = 15; xL = -0.004977; xH = 0.005299; x1 = -0.001552; x2 = 0.001874; p(x1) = 5.000005; p(x2)
iter = 16; xL = -0.004977; xH = 0.001874; x1 = -0.002693; x2 = -0.000410; p(x1) = 5.000014; p(x2)
iter = 17; xL = -0.002693; xH = 0.001874; x1 = -0.001171; x2 = 0.000352; p(x1) = 5.000003; p(x2)
iter = 18; xL = -0.001171; xH = 0.001874; x1 = -0.000156; x2 = 0.000859; p(x1) = 5.000000; p(x2)
iter = 19; xL = -0.001171; xH = 0.000859; x1 = -0.000494; x2 = 0.000182; p(x1) = 5.000000; p(x2)
iter = 20; xL = -0.000494; xH = 0.000859; x1 = -0.000043; x2 = 0.000408; p(x1) = 5.000000; p(x2)
iter = 21; xL = -0.000494; xH = 0.000408; x1 = -0.000194; x2 = 0.000107; p(x1) = 5.000000; p(x2)
iter = 22; xL = -0.000194; xH = 0.000408; x1 = 0.000007; x2 = 0.000207; p(x1) = 5.000000; p(x2) =
iter = 23; xL = -0.000194; xH = 0.000207; x1 = -0.000060; x2 = 0.000074; p(x1) = 5.000000; p(x2)
iter = 24; xL = -0.000194; xH = 0.000074; x1 = -0.000104; x2 = -0.000015; p(x1) = 5.000000; p(x2)
iter = 25; xL = -0.000104; xH = 0.000074; x1 = -0.000045; x2 = 0.000014; p(x1) = 5.000000; p(x2)
iter = 26; xL = -0.000045; xH = 0.000074; x1 = -0.000005; x2 = 0.000034; p(x1) = 5.000000; p(x2)
Local minimum point is (-0.000005, 5.000000)
```

```
$ ./a.out
1 2 0 5
-1.0 0.5
-1
iter = 1; xL = -1.000000; xH = 0.500000; x1 = -0.500000; x2 = 0.000000; p(x1) = 5.375000; p(x2) =
iter = 2; xL = -0.500000; xH = 0.500000; x1 = -0.166667; x2 = 0.166667; p(x1) = 5.050926; p(x2) =
iter = 3; xL = -0.500000; xH = 0.166667; x1 = -0.277778; x2 = -0.055556; p(x1) = 5.132888; p(x2)
iter = 4; xL = -0.277778; xH = 0.166667; x1 = -0.129630; x2 = 0.018519; p(x1) = 5.031429; p(x2) =
iter = 5; xL = -0.129630; xH = 0.166667; x1 = -0.030864; x2 = 0.067901; p(x1) = 5.001876; p(x2) =
iter = 6; xL = -0.129630; xH = 0.067901; x1 = -0.063786; x2 = 0.002058; p(x1) = 5.007878; p(x2) =
iter = 7; xL = -0.063786; xH = 0.067901; x1 = -0.019890; x2 = 0.024005; p(x1) = 5.000783; p(x2) =
iter = 8; xL = -0.063786; xH = 0.024005; x1 = -0.034522; x2 = -0.005258; p(x1) = 5.002342; p(x2)
iter = 9; xL = -0.034522; xH = 0.024005; x1 = -0.015013; x2 = 0.004496; p(x1) = 5.000447; p(x2) =
iter = 10; xL = -0.015013; xH = 0.024005; x1 = -0.002007; x2 = 0.010999; p(x1) = 5.000008; p(x2)
iter = 11; xL = -0.015013; xH = 0.010999; x1 = -0.006342; x2 = 0.002329; p(x1) = 5.000080; p(x2)
iter = 12; xL = -0.006342; xH = 0.010999; x1 = -0.000562; x2 = 0.005219; p(x1) = 5.000001; p(x2)
iter = 13; xL = -0.006342; xH = 0.005219; x1 = -0.002489; x2 = 0.001365; p(x1) = 5.000012; p(x2)
iter = 14; xL = -0.002489; xH = 0.005219; x1 = 0.000081; x2 = 0.002650; p(x1) = 5.000000; p(x2) =
iter = 15; xL = -0.002489; xH = 0.002650; x1 = -0.000776; x2 = 0.000937; p(x1) = 5.000001; p(x2)
iter = 16; xL = -0.002489; xH = 0.000937; x1 = -0.001347; x2 = -0.000205; p(x1) = 5.000004; p(x2)
iter = 17; xL = -0.001347; xH = 0.000937; x1 = -0.000585; x2 = 0.000176; p(x1) = 5.000001; p(x2)
iter = 18; xL = -0.000585; xH = 0.000937; x1 = -0.000078; x2 = 0.000429; p(x1) = 5.000000; p(x2)
iter = 19; xL = -0.000585; xH = 0.000429; x1 = -0.000247; x2 = 0.000091; p(x1) = 5.000000; p(x2)
iter = 20; xL = -0.000247; xH = 0.000429; x1 = -0.000022; x2 = 0.000204; p(x1) = 5.000000; p(x2)
iter = 21; xL = -0.000247; xH = 0.000204; x1 = -0.000097; x2 = 0.000054; p(x1) = 5.000000; p(x2)
iter = 22; xL = -0.000097; xH = 0.000204; x1 = 0.000003; x2 = 0.000104; p(x1) = 5.000000; p(x2) =
iter = 23; xL = -0.000097; xH = 0.000104; x1 = -0.000030; x2 = 0.000037; p(x1) = 5.000000; p(x2)
iter = 24; xL = -0.000097; xH = 0.000037; x1 = -0.000052; x2 = -0.000008; p(x1) = 5.000000; p(x2)
Local minimum point is (-0.000008, 5.000000)
```

```
$ ./a.out
0 1 1 -1
-1.0 1.0
```

```
-1
iter = 1; xL = -1.000000; xH = 1.000000; x1 = -0.333333; x2 = 0.333333; p(x1) = -1.222222; p(x2)
iter = 2; xL = -1.000000; xH = 0.333333; x1 = -0.555556; x2 = -0.111111; p(x1) = -1.246914; p(x2)
iter = 3; xL = -1.000000; xH = -0.111111; x1 = -0.703704; x2 = -0.407407; p(x1) = -1.208505; p(x2
iter = 4; xL = -0.703704; xH = -0.111111; x1 = -0.506173; x2 = -0.308642; p(x1) = -1.249962; p(x2
iter = 5; xL = -0.703704; xH = -0.308642; x1 = -0.572016; x2 = -0.440329; p(x1) = -1.244814; p(x2
iter = 6; xL = -0.572016; xH = -0.308642; x1 = -0.484225; x2 = -0.396433; p(x1) = -1.249751; p(x2
iter = 7; xL = -0.572016; xH = -0.396433; x1 = -0.513489; x2 = -0.454961; p(x1) = -1.249818; p(x2
iter = 8; xL = -0.572016; xH = -0.454961; x1 = -0.532998; x2 = -0.493980; p(x1) = -1.248911; p(x2
iter = 9; xL = -0.532998; xH = -0.454961; x1 = -0.506986; x2 = -0.480973; p(x1) = -1.249951; p(x2
iter = 10; xL = -0.532998; xH = -0.480973; x1 = -0.515656; x2 = -0.498315; p(x1) = -1.249755; p(x
iter = 11; xL = -0.515656; xH = -0.480973; x1 = -0.504095; x2 = -0.492534; p(x1) = -1.249983; p(x
iter = 12; xL = -0.515656; xH = -0.492534; x1 = -0.507949; x2 = -0.500242; p(x1) = -1.249937; p(x
iter = 13; xL = -0.507949; xH = -0.492534; x1 = -0.502811; x2 = -0.497673; p(x1) = -1.249992; p(x
iter = 14; xL = -0.502811; xH = -0.492534; x1 = -0.499385; x2 = -0.495960; p(x1) = -1.250000; p(x
iter = 15; xL = -0.502811; xH = -0.495960; x1 = -0.500527; x2 = -0.498244; p(x1) = -1.250000; p(x
iter = 16; xL = -0.502811; xH = -0.498244; x1 = -0.501288; x2 = -0.499766; p(x1) = -1.249998; p(x
iter = 17; xL = -0.501288; xH = -0.498244; x1 = -0.500274; x2 = -0.499259; p(x1) = -1.250000; p(x
iter = 18; xL = -0.501288; xH = -0.499259; x1 = -0.500612; x2 = -0.499935; p(x1) = -1.250000; p(x
iter = 19; xL = -0.500612; xH = -0.499259; x1 = -0.500161; x2 = -0.499710; p(x1) = -1.250000; p(x
iter = 20; xL = -0.500612; xH = -0.499710; x1 = -0.500311; x2 = -0.500010; p(x1) = -1.250000; p(x
iter = 21; xL = -0.500311; xH = -0.499710; x1 = -0.500111; x2 = -0.499910; p(x1) = -1.250000; p(x
iter = 22; xL = -0.500111; xH = -0.499710; x1 = -0.499977; x2 = -0.499843; p(x1) = -1.250000; p(x
iter = 23; xL = -0.500111; xH = -0.499843; x1 = -0.500022; x2 = -0.499932; p(x1) = -1.250000; p(x
iter = 24; xL = -0.500111; xH = -0.499932; x1 = -0.500051; x2 = -0.499992; p(x1) = -1.250000; p(x
iter = 25; xL = -0.500051; xH = -0.499932; x1 = -0.500012; x2 = -0.499972; p(x1) = -1.250000; p(x
Local minimum point is (-0.500012, -1.250000)
```

Click here to submit to CodeCrunch.

Check the correctness of the output by typing the following Unix command

```
./a.out < local.in | diff - local7.out
```

Submission (Course)

Select course: CS1010E (2017/2018 Sem 1) - Programming Methodology ▼

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.