THE MIDPOINT RULE

1. Table 1 below shows an algorithm for the midpoint rule for approximating integrals. The midpoint rule is

\[\int_a^b f(x) \, dx \approx [f(a + h/2) + f(a + 3h/2) + f(a + 5h/2) + \cdots + f(a + (n - 3/2)h) + f(b - h/2)] \cdot h\]

where \(h = (b - a)/n\) for \(n\) rectangles.

<table>
<thead>
<tr>
<th>GIVEN</th>
<th>the integrand (f)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>the limits (a) and (b)</td>
</tr>
<tr>
<td></td>
<td>the number (n) of rectangles</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STEP 1</th>
<th>compute (h = (b - a)/n)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>set theSum = 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STEP 2</th>
<th>loop: for (i) from 1 to (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>add (f(a + (i - 1/2) \cdot h)) to theSum</td>
</tr>
</tbody>
</table>

| STEP 3 | set result to the product of theSum and \(h\) |

| OUTPUT | return the value of result |

Table 1: A Midpoint Rule Algorithm

(a) Write statements implementing the midpoint rule in your choice of Maple, Matlab/Octave, C, or Java.

(b) Test the code by approximating \(\int_1^2 \frac{dx}{x}\) with \(n = 20\).

(Note: \(\int_1^2 \frac{dx}{x} = \ln(2) \approx 0.6931471806\) and \(M_{20} = 0.6930690983\) to 10 digits.)

Figure 1: Midpoint Rule Plot
Maple

\[ f := x -> 1/x; \]
\[ a, b := 1, 2; \]
\[ n := 20; \]
\[ h := (b-a)/n; \]
\[ \text{theSum} := 0; \]
\[ \text{for i from 1 to n do} \]
\[ \quad \text{theSum} := \text{theSum} + f(a + (i-1/2)*h); \]
\[ \text{end do}; \]
\[ \text{result} := \text{theSum} \ast h; \]
\[ \text{result}; \quad 0.6930690980 \]

Octave (Matlab)

\[ \text{function y = f(x)} \]
\[ \quad y = 1/x; \]
\[ \text{endfunction}; \]
\[ a = 1; \]
\[ b = 2; \]
\[ n = 20; \]
\[ h = (b-a)/n; \]
\[ \text{theSum} = 0; \]
\[ \text{for i = 1:n} \]
\[ \quad \text{theSum} = \text{theSum} + f(a + (i-1/2) \ast h); \]
\[ \text{endfor}; \]
\[ \text{result} = \text{theSum} \ast h; \]
\[ \text{octave-3.2.2:11> result} \]
\[ \text{result} = 0.69307 \]

C

wmcb% cat MidP.c
#include <stdio.h>
main() {
    float f(float x);
    float a, b, h, theSum, x, result;
    int n, i;
    a = 1.0;
    b = 2.0;
    n = 20;
    h = (b-a)/n;
    theSum = 0.0;
    for (i=1; i<=n; i++) {
        x = a+(i-0.5) \ast h;
        theSum = theSum + f(x);
    }
    result = theSum \ast h;
    printf("%f \n", result);
}

float f(float x) {
    float y;
    y = 1.0/x;
    return(y);
}

wmcb% gcc -o MidP MidP.c
wmcb% ./MidP
0.693069

Java

wmcb% cat MidP.java
class MidP {
    public static void main(String[] args) {
        float a, b, h, theSum, x, result;
        int n, i;
        a = 1.0f;
        b = 2.0f;
        n = 20;
        h = (b-a)/n;
        theSum = 0.0f;
        for (i=1; i<=n; i++) {
            x = a+(i-0.5f) \ast h;
            theSum = theSum + f(x);
        }
        result = theSum \ast h;
        System.out.println(result);
    }

    public static float f(float x) {
        float y;
        y = 1.0f/x;
        return(y);
    }
}

wmcb% javac MidP.java
wmcb% java MidP
0.69306916