

# wxMaxima for Calculus I: Errata, Bugs and Common Problems

## Contents

<b>1</b>	<b>Errata</b>	<b>2</b>
1.1	Module 0 . . . . .	2
1.2	Module 1 . . . . .	2
1.3	Module 2 . . . . .	2
1.4	Module 3 . . . . .	2
1.5	Module 4 . . . . .	2
1.6	Module 5 . . . . .	2
1.7	Module 6 . . . . .	2
<b>2</b>	<b>Bugs</b>	<b>3</b>
<b>3</b>	<b>Common Problems</b>	<b>4</b>

# 1 Errata

## 1.1 Module 0

1. pg. 19, Exercise 7: It is implied that the coordinates require a decimal approximation – they do not. In addition, “closed circles” should be written “closed points” (indicated with `point_type`).

## 1.2 Module 1

1. pg. 39, Example 1.6.2: In the statement of the Example, we are asked to define  $h(x)$  as the linear combination  $2f(x) + 3g(x)$ , but the entire problem is solved for  $h(x) = f(x) + g(x)$  instead (including the graph).

## 1.3 Module 2

1. pg. 56, Example 2.1.1: A previous assignment to `x` in a do-loop causes an error in `wxdraw2d`. `kill(all)$` should be executed, and the function  $f(x)$  redefined before graphing.
2. pg. 58, Example 2.1.2: The graph has the same issue as Example 2.1.1. Apply `kill(all)$`, and redefine the function  $f(x)$  to fix it.

## 1.4 Module 3

1. pg. 78, Example 3.3.2: In (%i7), the functions `POS(x)` and `NEG(x)` are called inside `diff`. These were, in fact, defined as *expressions* not functions, so we should refer to just `POS` and `NEG`.

## 1.5 Module 4

1. pg. 97, Example 4.4.1: `LINE_1` was assigned in (%i6) but is called as `LINE1` in (%i7). On the bottom of pg. 98, the same confusion occurs, where `LINE2` is used instead of `LINE_2` (this one doesn't cause an error, but it doesn't match the pattern for the rest of the example).

## 1.6 Module 5

## 1.7 Module 6

1. pg. 144, Exercise 11: The integrand is never negative – the idea is to divide the integral into sections for which  $v(t)$  is only positive or only negative.

## 2 Bugs

### 1. filled\_func:

This command requires the user to indicate an upper and lower function to shade between. In some versions of wxMaxima, `filled_func` malfunctions if the lower function is a constant. For example, the following code may produce a filled trapezoid rather than following the shape of  $f(x)$ :

```
filled_func=true,  
  filled_func=f(x),  
  explicit(0,x,-1,2),  
filled_func=false
```

If this bug causes a problem, you can work around it by adding a very small *explicit function of x*: `0.0000000001*x` (or any other sufficiently small number), for example:

```
filled_func=true,  
  filled_func=f(x),  
  explicit(0.0000000001*x,x,-1,2),  
filled_func=false
```

This idea can be extended to any other constant bounding function  $g(x) = c$  by just using `c+0.0000000000001*x`.

### 3 Common Problems

1. By default, **Enter** should create a line break, while **Shift+Enter** evaluates a command. If you want to change this configuration, go to Edit → Configure, and check or uncheck “Enter evaluates cells”.
2. Variable assignments persist until they are either reassigned or cleared. If you are bouncing around in a worksheet to work on several problems, `kill(all)$` can be used to clear all variable assignments at the beginning of each problem. Some users prefer to start every new problem this way “just in case”.
3. wxMaxima will frequently hang on bad code. If the program freezes or does not say “Ready for user input” in the lower right corner of the window, you may have to start a new wxMaxima session and copy/paste your code into that session.