**Binary 1:**  
For each binary, answer the below questions. Each question is worth 1 point.

* Using LordPE:
  + What is the address of the entry point?

0x401331

* + How big is the text section?

0x10993 Vsize (not sure what the RSize stands for… rounded? Maybe with padding and such.)

* + What are the starting and ending addresses of the .text section?

0x401000 start

0x411993 end

* Using X32DBG
  + What is the address of the entry point?

0x201331

* + What are the starting and ending addresses of the ‘main’ function?

0x201020 start

0x201060 end

* + What calling convention is used?

cdecl

**Binary 2:**  
For each binary, answer the below questions. Each question is worth 1 point.

* Using LordPE:
  + What is the address of the entry point?

0x4013A1

* + How big is the text section?

0x10CB3

* + What are the starting and ending addresses of the .text section?

0x401000 start

0x411CB3 end

* Using X32DBG
  + What is the address of the entry point?

0x4313A1

* + What are the starting and ending addresses of the ‘main’ function?

0x431080 start

0x4310D0 end

* + What calling convention is used?

cdecl

**Binary 3:**  
For each binary, answer the below questions. Each question is worth 1 point.

* Using LordPE:
  + What is the address of the entry point?

0x4013F2

* + How big is the text section?

0x10CF3

* + What are the starting and ending addresses of the .text section?

0x401000 start

0x411CF3 end

* Using X32DBG
  + What is the address of the entry point?

0x6413F2

* + What are the starting and ending addresses of the ‘main’ function?

0x641080 start

This function’s ret call is located at 0x641111, however it never reaches that point because the final function called in main calls an int 0x29 (\_\_fastfail) at address 0x0064143B which exits the program before main can return

From what I see crawling around inside that function the function that calls \_\_fastfail looks like a system function. It’s checking processor features, and process IDs and all that jazz.

* + What calling convention is used?

cdecl