|  |  |
| --- | --- |
| Camille Bordes |  |

EFREI CA2023F

Computer Architecture – Assembly language programming 4

Answer sheet

# Determining size of .float data type

Analyze the source code available in **fpu1.s** file. do necessary modifications that allows you to print on screen size of .float type variable. Then write down into a frame the size of .float

The size of float variable in x86 architecture is:

|  |
| --- |
| 32-bit (4-bytes) |

# Determining size of .double data type

Analyze the source code available in **fpu1.s** file. do necessary modifications that allows you to print on screen size of .double type variable. Then write down into a frame the size of .double.

The size of double variable in x86 architecture is:

|  |
| --- |
| 64-bit (8-bytes) |

# Representation of double type number

In data section of the source code **fpu1.s** there is a variable user\_double . This variable is initialized by value 4.17. Change that value do the different non-integer number.

assembly and execute the program.

Write down (into the frame) your value and its hexadecimal representation:

Your .double value:

|  |
| --- |
| 12.3 |

Hexadecimal representation of your .double value:

|  |
| --- |
| 40289999 9999999a |

Explain in details how this hexadecimal number can be interpreted as your non-integer value.

|  |
| --- |
| The hexadecimal representation 402599999999999a can be interpreted as the non-integer value 12.3 in double precision format according to the IEEE 754 standard. The sign bit is 0, indicating a positive value, and the exponent bits are 10000000001, representing an exponent of 1025 after applying the bias of 1023. The fraction bits are 1.0110011001100110011001100110011001100110011001100, representing the binary fraction 1.0110011001100110011001100110011001100110011001100. By combining these components, we conclude that the hexadecimal representation 402599999999999a represents the non-integer value 12.3 in double precision format. |

# Representation of float number

In data section of the source code **fpu1.s** there is a variable *user\_float* . This variable is initialized by value 4.17. Change that value do the different non-integer number.

Modify fpu1 program in such a way that it displays the hexadecimal representation of the *user\_float* value and run it

Write down (into the frame) your value and its hexadecimal representation:

Your value:

|  |
| --- |
| 12.3 |

Hexadecimal representation of your value:

|  |
| --- |
| 4144cccd |

Explain in details how this hexadecimal number can be interpreted as your non-integer value.

|  |
| --- |
| The hexadecimal number 4144CCCD can be interpreted as the non-integer value 12.3 in single precision float format (float) using the IEEE 754 standard. The sign bit is 0, indicating a positive value. The exponent bits are 10000010, representing an exponent of 130 after applying the bias of 127. The fraction bits are 1.100110011001100110011, representing the binary fraction 1.100110011001100110011. Combining these components, the hexadecimal representation 4144CCCD represents the non-integer value 12.3 in single precision float format. |