

Solution

Problem 1

[1]	1111 1111	[2]	0100
[3]	1000	[4]	1111 1110
[5]	0011 0011	[6]	0000 0001

Problem 2

[1]	0x7e5	[2]	0x2000000e
[3]	0x1ffffffff ffffffff	[4]	0x6000002a5
[5]	0x2018	[6]	0x10203
[7]	0x7e5	[8]	0x1ffffffff ffffffff
[9]	0x123456789abcdef	[10]	0xffffffff ffffffff

Problem 3

1. [1]	31	[2]	0x7e00
[3]	0x8001	[4]	0x7dff
2. 0xc3f2			
sign:0x1	exp:0x21	frac:0x1f2	

Problem 4

1. [1]	8	[2]	8
[3]	0x550000001058	[4]	0x550000001050
[5]	0x550000001050	[6]	0x550000001050
[7]	0x550000001048	[8]	0x550000001040
2.	$32 - (4+8+8+1) = 11$ Or $4+7=11$		
3.	3 bytes are wasted. Place "char c" after "int flags".		

Problem 5

1 [1]	case 38:	[2]	ret = *yp;
[3]	case 39:	[4]	5
[5]	.L3	[6]	.L8
[7]	.L4	[8]	.L6
2 [1]	0x87654388	[2]	0x87654388
[3]	0x88774321	[4]	0x0

Problem 6:

- 1 [1]: Pass `y` as the first argument to function `P`.
[2]: Save return value of `P` as `%rax` will be modified by second call to `Q` in line 10.
- 2 No. Because `%r11` is caller saved register, so it might be changed by invocation of `Q` in line 7, however `P` use it in line 9 without saving its value in line 5.
- 3 No. Because `%r12` is callee saved register, but `P` does not preserve its old value (unless `%rbp` in line 2 and line 14 are also changed to `%r12`).