

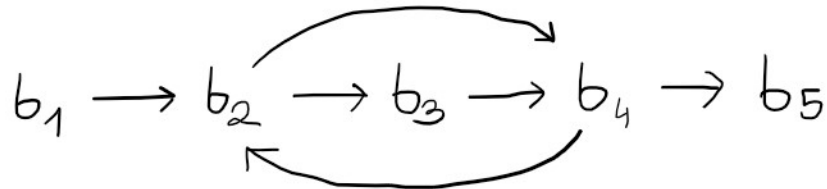
## Zad 5.

sobota, 1 kwietnia 2023 14:25

**Zadanie 5 (2).** Poniżej widnieje kod funkcji o sygnaturze «uint32\_t puzzle3(uint32\_t n, uint32\_t d)». Wyznacz bloki podstawowe oraz narysuj graf przepływu sterowania, po czym przetłumacz tę funkcję na język C. Na podstawie ustępu „*Mixing C and Assembly Language*” strony [GNU Assembler Examples<sup>2</sup>](#) napisz i zaprezentuj działanie programu, który pomógł Ci odpowiedzieć na pytanie co ta funkcja robi.

<pre>1 puzzle3: 2     movl %edi, %edi 3     salq \$32, %rsi 4     movl \$32, %edx 5     movl \$0x80000000, %ecx 6     xorl %eax, %eax 7 .L3: addq %rdi, %rdi 8     movq %rdi, %r8 9     subq %rsi, %r8 10    js .L2</pre>	$b_1$	<pre>11     orl %ecx, %eax 12     movq %r8, %rdi 13 .L2: shr1 %ecx 14     decl %edx 15     jne .L3 16     ret</pre>	$b_3$ $b_4$ $b_5$
---	-------	---	-------------------------

$b_2$



```
23 // ???
24 uint32_t puzzle3(uint32_t n, uint32_t d) {
25     uint64_t new_n = n;
26     uint64_t new_d = d <= 32;
27
28     uint32_t bit_pointer = 0x80000000;
29     uint32_t result = 0;
30
31     for (int i = 0; i < 32; i++) {
32         new_n <= 1;
33         if (new_n - new_d >= 0) {
34             result |= bit_pointer;
35             new_n = new_n - new_d;
36         }
37         bit_pointer >>= 1;
38     }
39
40     return result;
41 }
```

```

C test.c > ...
1  #include <stdio.h>
2  #include <inttypes.h>
3
4
5  // https://stackoverflow.com/questions/9280654/c-printing-bits
6  #define print_bits(x) \
7      do { \
8          unsigned long long a__ = (x); \
9          size_t bits__ = sizeof(x) * 8; \
10         printf("#x ": "); \
11         while (bits__--) putchar(a__ &(1ULL << bits__) ? '1' : '0'); \
12         putchar('\n'); \
13     } while (0)
14
15     uint32_t puzzle(uint32_t n, uint32_t d);
16
17     int main() {
18         uint32_t x = puzzle(10, 5);
19         printf("%u\n", x);
20         print_bits(x);
21     }

```

wychodzi na to, że to robi  $\lfloor \frac{n}{d} \rfloor$