

# Architektury systemów komputerowych

## Lista zadań nr 5

Na zajęcia 1–2 kwietnia 2019

**UWAGA!** Rozwiązania muszą się stosować do wytycznych podanych w nagłówku poprzedniej listy zadań!

**Zadanie 1.** Zaimplementuj funkcję zdefiniowaną poniżej w asemblerze x86-64. Taka procedura w języku C miałaby sygnaturę «long cmp(uint64\_t x, uint64\_t y)».

$$\text{cmp}(x, y) = \begin{cases} -1 & \text{gdy } x < y \\ 1 & \text{gdy } x > y \\ 0 & \text{gdy } x = y \end{cases}$$

**Wskazówka:** Rozwiązanie wzorcowe ma cztery wiersze (bez ret). Użyj instrukcji adc, sbb i neg.

**Zadanie 2.** Poniżej zamieszczono kod procedury o sygnaturze «long puzzle2(char \*s, char \*d)». Wyznacz **bloki podstawowe** oraz narysuj **graf przepływu sterowania**. Przetłumacz tę procedurę na język C, a następnie jednym zdaniem powiedz co ona robi.

```
1 puzzle2:                                10      cmpb  %c1, %r9b
2      movq  %rdi, %rax                    11      jne   .L2
3  .L3:  movb  (%rax), %r9b                 12      movq  %r8, %rax
4      leaq  1(%rax), %r8                  13      jmp   .L3
5      movq  %rsi, %rdx                    14  .L4:  subq  %rdi, %rax
6  .L2:  movb  (%rdx), %c1                  15      ret
7      incq  %rdx
8      testb %c1, %c1
9      je   .L4
```

**Zadanie 3.** 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](http://cs.lmu.edu/~ray/notes/gasexamples/)<sup>1</sup> napisz program, który pomoże Ci powiedzieć co ta funkcja robi.

```
1 puzzle3:                                11      orl   %ecx, %eax
2      movl  %edi, %edi                    12      movq  %r8, %rdi
3      salq  $32, %rsi                     13  .L2:  shrl  %ecx
4      movl  $32, %edx                     14      decl  %edx
5      movl  $0x80000000, %ecx             15      jne   .L3
6      xorl  %eax, %eax                    16      ret
7  .L3:  addq  %rdi, %rdi
8      movq  %rdi, %r8
9      subq  %rsi, %r8
10     js   .L2
```

<sup>1</sup><http://cs.lmu.edu/~ray/notes/gasexamples/>

**Zadanie 4.** Poniżej zamieszczono kod rekurencyjnej procedury o sygnaturze «int puzzle4(long \*a, long v, uint64\_t s, uint64\_t e)». Wyznacz bloki podstawowe oraz narysuj graf przepływu sterowania. Przetłumacz tę procedurę na język C, a następnie jednym zdaniem powiedz co ona robi.

```

1 puzzle4:                                11      cmpq  %rsi, %r8
2      movq  %rcx, %rax                    12      jg    .L11
3      subq  %rdx, %rax                    13      leaq  1(%rax), %rdx
4      shrq  %rax                          14      call  puzzle4
5      addq  %rdx, %rax                    15  .L10:  ret
6      cmpq  %rdx, %rcx                    16  .L11:  leaq  -1(%rax), %rcx
7      jb   .L5                            17      call  puzzle4
8      movq  (%rdi,%rax,8), %r8            18      ret
9      cmpq  %rsi, %r8                    19  .L5:  movl  $-1, %eax
10     je   .L10                            20      ret

```

**Wskazówka:** Z reguły procedurę «puzzle4» woła się następująco: «i = puzzle4(a, v, 0, n - 1)».

**Zadanie 5.** Poniżej widnieje kod procedury o sygnaturze «long puzzle5(void)». Podaj rozmiar i składowe rekordu aktywacji procedury «puzzle5». Jaką sygnaturę ma procedura «readlong»? Przetłumacz procedurę «puzzle5» na język C i wytłumacz jednym zdaniem co ona robi.

```

1 puzzle5:                                8      cqto
2      subq  $24, %rsp                      9      idivq 8(%rsp)
3      movq  %rsp, %rdi                    10     xorl  %eax, %eax
4      call  readlong                      11     testq %rdx, %rdx
5      leaq  8(%rsp), %rdi                 12     sete  %al
6      call  readlong                      13     addq  $24, %rsp
7      movq  (%rsp), %rax                  14     ret

```

**Zadanie 6.** Poniższy kod w asemblerze otrzymano w wyniku deasemblacji funkcji zadeklarowanej jako «long switch\_prob(long x, long n)». Zapisz w języku C kod odpowiadający tej funkcji.

```

1 400590 <switch_prob>:
2 400590: 48 83          subq  $0x3c,%rsi
3 400594: 48 83 fe 05    cmpq  $0x5,%rsi
4 400598: 77 29         ja   *0x4005c3
5 40059a: ff 24 f5 f8 06 40 00 jmpq  *0x4006f8(,%rsi,8)
6 4005a1: 48 8d 04 fd 00 00 00 00 lea  0x0(,%rdi,8),%rax
7 4005a9: c3          retq
8 4005aa: 48 89 f8     movq  %rdi,%rax
9 4005ad: 48 c1 f8 03   sarq  $0x3,%rax
10 4005b1: c3         retq
11 4005b2: 48 89 f8     movq  %rdi,%rax
12 4005b5: 48 c1 e0 04   shlq  $0x4,%rax
13 4005b9: 48 29 f8     subq  %rdi,%rax
14 4005bc: 48 89 c7     movq  %rax,%rdi
15 4005bf: 48 0f af ff   imulq %rdi,%rdi
16 4005c3: 48 8d 47 4b   leaq  0x4b(%rdi),%rax
17 4005c7: c3         retq

```

Zrzut pamięci przechowującej  
tablicę skoków:

```

18 (gdb) x/6gx 0x4006f8
19 0x4006f8: 0x4005a1
20 0x400700: 0x4005a1
21 0x400708: 0x4005b2
22 0x400710: 0x4005c3
23 0x400718: 0x4005aa
24 0x400720: 0x4005bf

```

**Zadanie 7.** Procedurę ze zmienną liczbą parametrów używającą pliku nagłówkowego `stdarg.h`<sup>2</sup> skompilowano z opcjami «-Og -mno-sse». Po jej deasemblacji otrzymano następujący wydruk. Co robi ta procedura i jaka jest jej sygnatura? Jakie dane są przechowywane w rekordzie aktywacji tej procedury?

```

1 puzzle7:
2     movq %rsi, -40(%rsp)
3     movq %rdx, -32(%rsp)
4     movq %rcx, -24(%rsp)
5     movq %r8, -16(%rsp)
6     movq %r9, -8(%rsp)
7     movl $8, -72(%rsp)
8     leaq 8(%rsp), %rax
9     movq %rax, -64(%rsp)
10    leaq -48(%rsp), %rax
11    movq %rax, -56(%rsp)
12    movl $0, %eax
13    jmp .L2
14    .L3: movq -64(%rsp), %rdx
15        leaq 8(%rdx), %rcx
16        movq %rcx, -64(%rsp)
17    .L4: addq (%rdx), %rax
18    .L2: subq $1, %rdi
19        js .L6
20        cmpl $47, -72(%rsp)
21        ja .L3
22        movl -72(%rsp), %edx
23        addq -56(%rsp), %rdx
24        addl $8, -72(%rsp)
25        jmp .L4
26    .L6: ret

```

**Wskazówka:** Przeczytaj rozdział §3.5.7 dokumentu opisującego ABI dostępnego na stronie przedmiotu.

**Zadanie 8.** Poniżej zamieszczono kod procedury o sygnaturze «struct T puzzle8(long \*a, long n)». Na jego podstawie podaj definicję typu «struct T». Przetłumacz tę procedurę na język C, po czym jednym zdaniem powiedz co ona robi.

```

1 puzzle8:
2     movq %rdx, %r11
3     xorl %r10d, %r10d
4     xorl %eax, %eax
5     movq $LONG_MIN, %r8
6     movq $LONG_MAX, %r9
7    .L2: cmpq %r11, %r10
8        jge .L5
9        movq (%rsi,%r10,8), %rcx
10       cmpq %rcx, %r9
11       cmovg %rcx, %r9
12       cmpq %rcx, %r8
13       cmovl %rcx, %r8
14       addq %rcx, %rax
15       incq %r10
16       jmp .L2
17    .L5: cqto
18        movq %r9, (%rdi)
19        idivq %r11
20        movq %r8, 8(%rdi)
21        movq %rax, 16(%rdi)
22        movq %rdi, %rax
23        ret

```

**Wskazówka:** Zauważ, że wynik procedury nie mieści się w rejestrach %rax i %rdx.

<sup>2</sup><https://en.wikipedia.org/wiki/Stdarg.h>