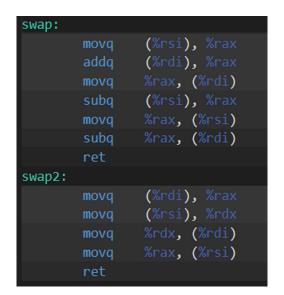
**Zadanie 1.** Intencją procedury «swap» jest zamiana wartości przechowywanych w komórkach pamięci o adresie «xp» i «yp». Odwołując się do pojęcia **aliasingu pamięci** (ang. *memory aliasing*) wytłumacz czemu kompilator nie może zoptymalizować poniższej procedury do procedury «swap2»? Pomóż mu zoptymalizować «swap» posługując się słowem kluczowym «restrict» i wyjaśnij jego znaczenie.

```
1 void swap(long *xp, long *yp) {
2  *xp = *xp + *yp; /* x+y */
3  *yp = *xp - *yp; /* x+y-y = x */
4  *xp = *xp - *yp; /* x+y-x = y */
5 }

1 void swap(long *xp, long *yp) {
2  long x = *xp, y = *yp;
3  x = x + y, y = x - y, x = x - y;
4  *xp = *xp - *yp; /* x+y-x = y */
5 }
```

kompilator nie ma pewności czy coś innego nie korzysta z wartości pod adresami xp, yp



```
swap:

movq (%rsi), %rax
movq (%rdi), %rdx
movq %rax, (%rdi)
movq %rdx, (%rsi)
ret

t restrict
```

informacja dla

kompilatora, że nic innego
nie koizystą z danych
zasobow