操作系统第 15 章作业

1. 用种子 1、2 和 3 运行,并计算进程生成的每个虚拟地址是处于界限内还是界限外?如果在界限内,请计算地址转换。

随机种子为 1：

```
yangjie@ubuntu:~/share/Operating-Systems-Three-Easy-Pieces-NOTES-main/15.第十五
章-机制_地址转换$ python2 relocation.py -s 1

ARG seed 1
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base    : 0x0000363c (decimal 13884)
  Limit   : 290

Virtual Address Trace
  VA  0: 0x0000030e (decimal:  782) --> PA or segmentation violation?
  VA  1: 0x00000105 (decimal:  261) --> PA or segmentation violation?
  VA  2: 0x000001fb (decimal:  507) --> PA or segmentation violation?
  VA  3: 0x000001cc (decimal:  460) --> PA or segmentation violation?
  VA  4: 0x0000029b (decimal:  667) --> PA or segmentation violation?

For each virtual address, either write down the physical address it translates to
OR write down that it is an out-of-bounds address (a segmentation violation). For
this problem, you should assume a simple virtual address space of a given size.
```

782>290，超出界限寄存器范围，非法地址

261<290，合法地址，PA=VA+Base=14145(0x00003741)

507>290，超出界限寄存器范围，非法地址

460>290，超出界限寄存器范围，非法地址

667>290，超出界限寄存器范围，非法地址

添加-c 参数：

```
章-机制_地址转换$ python2 relocation.py -s 1 -c

ARG seed 1
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base    : 0x0000363c (decimal 13884)
  Limit   : 290

Virtual Address Trace
  VA  0: 0x0000030e (decimal:  782) --> SEGMENTATION VIOLATION
  VA  1: 0x00000105 (decimal:  261) --> VALID: 0x00003741 (decimal: 14145)
  VA  2: 0x000001fb (decimal:  507) --> SEGMENTATION VIOLATION
  VA  3: 0x000001cc (decimal:  460) --> SEGMENTATION VIOLATION
  VA  4: 0x0000029b (decimal:  667) --> SEGMENTATION VIOLATION
```

经验证正确。

随机种子为 2：

```
yangjie@ubuntu:~/share/Operating-Systems-Three-Easy-Pieces-NOTES-main/15.第十五章-
机制_地址转换$ python2 relocation.py -s 2

ARG seed 2
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base    : 0x00003ca9 (decimal 15529)
  Limit   : 500

Virtual Address Trace
  VA  0: 0x00000039 (decimal:    57) --> PA or segmentation violation?
  VA  1: 0x00000056 (decimal:    86) --> PA or segmentation violation?
  VA  2: 0x00000357 (decimal:   855) --> PA or segmentation violation?
  VA  3: 0x000002f1 (decimal:   753) --> PA or segmentation violation?
  VA  4: 0x000002ad (decimal:   685) --> PA or segmentation violation?

For each virtual address, either write down the physical address it translates to
OR write down that it is an out-of-bounds address (a segmentation violation). For
this problem, you should assume a simple virtual address space of a given size.
```

57<500，合法地址，PA=VA+Base=15586(0x00003ce2)

86<500，合法地址，PA=VA+Base=15615(0x00003cff)

855>500，超出界限寄存器范围，非法地址

753>500，超出界限寄存器范围，非法地址

685>500，超出界限寄存器范围，非法地址

添加-c 参数：

```
yangjie@ubuntu:~/share/Operating-Systems-Three-Easy-Pieces-NOTES-main/15.第十五章-
机制_地址转换$ python2 relocation.py -s 2 -c

ARG seed 2
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base    : 0x00003ca9 (decimal 15529)
  Limit   : 500

Virtual Address Trace
  VA  0: 0x00000039 (decimal:    57) --> VALID: 0x00003ce2 (decimal: 15586)
  VA  1: 0x00000056 (decimal:    86) --> VALID: 0x00003cff (decimal: 15615)
  VA  2: 0x00000357 (decimal:   855) --> SEGMENTATION VIOLATION
  VA  3: 0x000002f1 (decimal:   753) --> SEGMENTATION VIOLATION
  VA  4: 0x000002ad (decimal:   685) --> SEGMENTATION VIOLATION
```

经验证正确。

随机种子为 3：

```
机制_地址转换$ python2 relocation.py -s 3

ARG seed 3
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base   : 0x000022d4 (decimal 8916)
  Limit  : 316

Virtual Address Trace
  VA  0: 0x0000017a (decimal:  378) --> PA or segmentation violation?
  VA  1: 0x0000026a (decimal:  618) --> PA or segmentation violation?
  VA  2: 0x00000280 (decimal:  640) --> PA or segmentation violation?
  VA  3: 0x00000043 (decimal:   67) --> PA or segmentation violation?
  VA  4: 0x0000000d (decimal:   13) --> PA or segmentation violation?

For each virtual address, either write down the physical address it translates to
OR write down that it is an out-of-bounds address (a segmentation violation). For
this problem, you should assume a simple virtual address space of a given size.
```

378>316，超出界限寄存器范围，非法地址

618>316，超出界限寄存器范围，非法地址

640>316，超出界限寄存器范围，非法地址

67<316，合法地址，PA=VA+Base=8983(0x00002317)

13<316，合法地址，PA=VA+Base=8929(0x000022e1)

添加-c 参数：

```
yangjie@ubuntu:~/share/Operating-Systems-Three-Easy-Pieces-NOTES-main/15.第十五章-
机制_地址转换$ python2 relocation.py -s 3 -c

ARG seed 3
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base   : 0x000022d4 (decimal 8916)
  Limit  : 316

Virtual Address Trace
  VA  0: 0x0000017a (decimal:  378) --> SEGMENTATION VIOLATION
  VA  1: 0x0000026a (decimal:  618) --> SEGMENTATION VIOLATION
  VA  2: 0x00000280 (decimal:  640) --> SEGMENTATION VIOLATION
  VA  3: 0x00000043 (decimal:   67) --> VALID: 0x00002317 (decimal: 8983)
  VA  4: 0x0000000d (decimal:   13) --> VALID: 0x000022e1 (decimal: 8929)
```

经验证正确。

2. 使用以下标志运行:-s 1 -n 10 -l 100。可以设置界限的最大值是多少,以便地址空间仍然完全放在物理内存中?

因为已经通过-l 参数设置了界限寄存器的值，所以这里问的应该是基址寄存器的最大值（查阅英文版原教材，也确实是这样）

```
yangjie@ubuntu:~/share/Operating-Systems-Three-Easy-Pieces-NOTES-main/15.第十五章-机制_地址转换$ python2 relocation.py -s 1 -n 10 -l 100

ARG seed 1
ARG address space size 1k
ARG phys mem size 16k

Base-and-Bounds register information:

  Base   : 0x00000899 (decimal 2201)
  Limit  : 100

Virtual Address Trace
  VA  0: 0x00000363 (decimal:  867) --> PA or segmentation violation?
  VA  1: 0x0000030e (decimal:  782) --> PA or segmentation violation?
  VA  2: 0x00000105 (decimal:  261) --> PA or segmentation violation?
  VA  3: 0x000001fb (decimal:  507) --> PA or segmentation violation?
  VA  4: 0x000001cc (decimal:  460) --> PA or segmentation violation?
  VA  5: 0x0000029b (decimal:  667) --> PA or segmentation violation?
  VA  6: 0x00000327 (decimal:  807) --> PA or segmentation violation?
  VA  7: 0x00000060 (decimal:   96) --> PA or segmentation violation?
  VA  8: 0x0000001d (decimal:   29) --> PA or segmentation violation?
  VA  9: 0x00000357 (decimal:  855) --> PA or segmentation violation?

For each virtual address, either write down the physical address it translates to
OR write down that it is an out-of-bounds address (a segmentation violation). For
this problem, you should assume a simple virtual address space of a given size.
```

物理内存大小为 16*1024 比特，要使地址空间完全放在物理内存中，基址寄存器最大值为 16*1024-100=16284.