Operating System **Hsu PO-CHUN**
10610 CS342302 Student ID: s105062803
Prof. Pai H. Chou Submitted: Dec. 12, 2017
TA: Due Date: Dec. 12, 2017

Assignment 8

1 Problem 1

[10 points] 8.15 Consider a logical address space of 256 pages, with a 4KB page size, mapped onto a physical memory of 64 frames.

a. How many bits are required in the logical address?

b. How many bits are required in the physical address?

1.1 Answer

a. In the logical address, it requires $lg(256\ pages * 4K\ page\ size)$ = $lg(2^{20})$ = 20 bits.

b. In the physical address, it requires $lg(64\ frames * 4K\ frames\ size)$ = $lg(2^{18})$ = 18 bits.

2 Problem 2

[20 points] 8.16 Consider a computer system with a 32-bit logical address and 4-KB page size. The system supports up to 512-MB of physical memory. How many entries are there in each of the following?

a. A conventional single-level page table

b. An inverted page table

2.1 Answer

a. A conventional single-level page table has $\frac{2^{32}}{4K}$ = $\frac{2^{32}}{2^{12}}$ = 2^{20} entries.

b. An inverted page table has $\frac{512M}{4K}$ = $\frac{2^{29}}{2^{12}}$ = 2^{17} entries.

3 Problem 3

[10 points] 8.20 Consider the following segment table:

Table 1:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?

a. 0, 430 (segment#, logical address)

b. 1, 10

c. 2, 500

d. 3, 4000

e. 4, 112

3.1 Answer

a. (d=430 $<$ limit=600) So, the physical address are: 219 + 430 = 649

b. (d=10 $<$ limit=14) So, the physical address are: 2300 + 10 = 2310

c. (d=500 $\not<$ limit=100) So, a trap:addressing error to OS

d. (d=4000 $\not<$ limit=580) So, a trap:addressing error to OS

e. (d=112 $\not<$ limit=96) So, a trap:addressing error to OS

4 Bonus

Test case request:

```
requests =
[('a', 9), ('b', 7), ('c', 1), ('d', 3),# malloc
 ('d', None), ('b', None),# free
 ('f', 2),# best fit (17, 2), first fit (9, 2), worst fit (9, 2)
 ('g', 3),# best fit (9, 3), first fit (11, 3), worst fit (11, 3)
 ('h', 2),# best fit (12, 2), first fit (14, 2), worst fit (17, 2)
 ('i',3)# both best fit and worst fit fail, but first fit succeeds
]
```

Explain:
After the request tuple ('b',None), the hole:(9,7) and (17,3) are produced.
The request tuple ('f',2):
Best Fit will allocate at (17,2),and both first fit and worst fit will allocate (9,2) because the remind size of the holes is 1, which is the smallest.

The request tuple ('g',3):
Best Fit will allocate at (9,3) for the only available hole,and both first fit and worst fit will allocate (11,3) for the first available hole and the largest reminding size 5 respectively.

The request tuple ('h',2):
Best Fit will allocate at (12,2) for the only available hole,and the first fit will allocate at (14,2) for the first available hole. And the worst fit will allocated at (17,2) for the reminding size is 1, which is the largest.

The request tuple ('i',3):
Both best Fit and worst fit fail to be allocated, but first fit succeeds at (17,3).