Karlsruhe Institute of Technology

Operating Systems 2016/17
Tutorial-Assignment 11

Prof. Dr. Frank Bellosa
Dipl.-Inform. Marc Rittinghaus

Question 11.1: Hardware- vs. Software-Walked Page Tables

a. What's the difference between the use of hardware-walked page tables and software-walked page tables? How does this relate to TLBs?

b. What difference can you make out in the contents of software-walked vs. hardware-walked multi-level page tables?

c. Under what circumstances are TLB miss handlers or page fault handlers invoked?

Question 11.2: Page Fault Handling

a. Explain the terms demand-paging and pre-paging. What are the respective strengths and weaknesses?

b. When a thread touches a page for the first time with demand-paging, a page fault will occur. Classify the page fault according to where the data for the unmapped page has to be fetched from by the page fault handler.

c. If the process has been running for some time, modifying data along its way, there is one additional case that needs to be covered on a page fault. Which?

d. Discuss which information is required by the page fault handler to correctly setup (or restore) the contents of accessed pages.

e. Can you reuse page table entries to store some of this information? Is it a good idea?

f. What is Copy-on-Write? How can it be implemented?

g. Recap: Describe the steps necessary to handle a page fault in an application's address space.

Question 11.3: Page Replacement Basics

a. The pager of some systems tries to always offer a certain amount of free page frames to improve paging. What is the basic idea behind such a pager?

b. Describe the difference between a global and a local page replacement algorithm. Discuss the advantages and disadvantages of each of them.

c. Does a virtual memory system implementing equal allocation require a global or a local page replacement policy? Justify your answer.

d. What is thrashing? When does it occur?

e. What is the working set of a process? How can the working set be used to prevent thrashing?

Question 11.4: Page Replacement Policies

a. A task has four page frames $(0, \ldots, 3)$ allocated to it. The virtual page number of each page frame, the time of the last loading of a page into each page frame, the time of the last access to the page frame, and the referenced (R) and modified (M) bits of each page frame are shown in the following table.

frame	virtual page	load time	access time	referenced	modified
0	2	60	161	0	1
1	1	130	160	0	0
2	0	26	162	1	0
3	3	20	163	1	1

A pagefault to virtual page 4 occurs. Which page frame will have its contents replaced for the *FIFO*, *LRU*, *Clock* and *Optimal* (with respect to the number of page replacements) replacement policies?

For the Clock algorithm assume that the circular buffer is ordered ascending by load time and that the next-frame pointer refers to frame 3.

For the Optimal algorithm use the following string for subsequent references:
4, 0, 0, 0, 2, 4, 2, 1, 0, 3, 2.

Explain the reason in each case.

b. Evaluate *stack*, *code*, and *heap* as to how well you expect the LRU page replacement policy to perform on them. Explain your opinion for each segment.