CS 240
Computer Systems

#5: Locality of Reference and Virtual Memory

Feb. 1, 2022 · Wade Fagen-Ulmschneider

Sample Programs:

04cr.c

```c
for (unsigned int c = 0; c < SIZE; c++) {
  for (unsigned int r = 0; r < SIZE; r++) {
    array[(r * SIZE) + c] = (r * SIZE) + c;
  }
}
```

04rc.c

```c
for (unsigned int r = 0; r < SIZE; r++) {
  for (unsigned int c = 0; c < SIZE; c++) {
    array[(r * SIZE) + c] = (r * SIZE) + c;
  }
}
```

Running Times: `04cr.c` (Program #1):

`04rc.c` (Program #2):

Caching Strategies: Keeping Data Close

In working with memory in any computer system, we want to access it as quickly as possible. However, space is extremely limited in the fastest memory, so we need strategies on what data to keep close. General Purpose Memory:

- CPU Registers:
- CPU Cache (i7-12700K, Released Q4'21):
- RAM:

Key Idea: Locality of Reference

System Memory: Limited, Shared, and Simple

1.
2.
3.

To help us to begin to organize this RAM, we divide the RAM up into chunks called __________.

On Linux, find the size of a page:

```
# getconf PAGESIZE
```

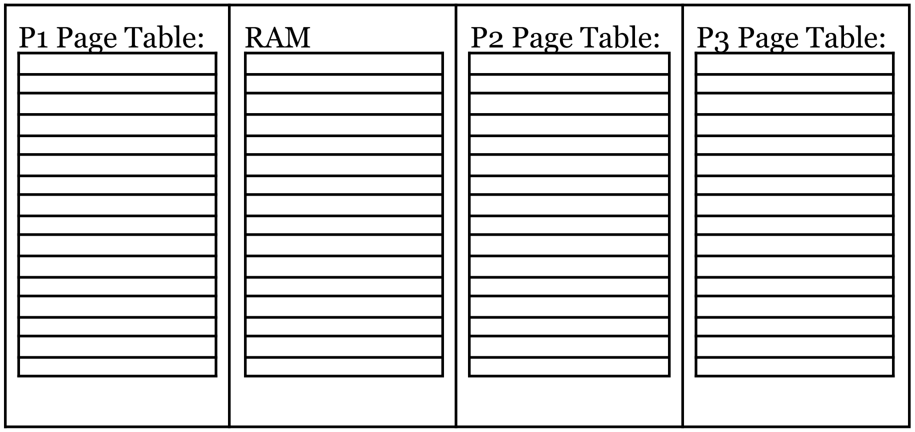
- On almost every modern system, a page is ______ KB.

Virtual Memory:

Modern systems provide an abstraction between the _________ and ____________________:

1. A ____________________ translates a ____________ into a physical address.
2. Every memory address is made up of the _____________ and the _____________:
3. Virtual Memory is **NOT shared** between processes/apps:
4. **EVERY memory address** you have ever seen is a virtual memory address!

Let's explore a sequence of allocations using a page table:



Allocation Sequence:

1. Process #1 (P1): **`a = malloc(3 * 4096)`**
2. Process #3 (P3): **`b = malloc(5 * 4096)`**
3. Process #1 (P1): **`c = malloc(2 * 4096)`**
4. Process #3 (P3) exits.
5. Process #2 (P2): **`d = malloc(4 * 4096)`**
6. Process #2 (P2): **`e = malloc(5 * 4096)`**
7. Process #1 (P1): **`a = realloc(a, 5 * 4096)`**

With a virtual memory system:

- Can we meet all of the allocation requests?
- Are we limited to just RAM?

Advantages of a Virtual Memory System:

1.

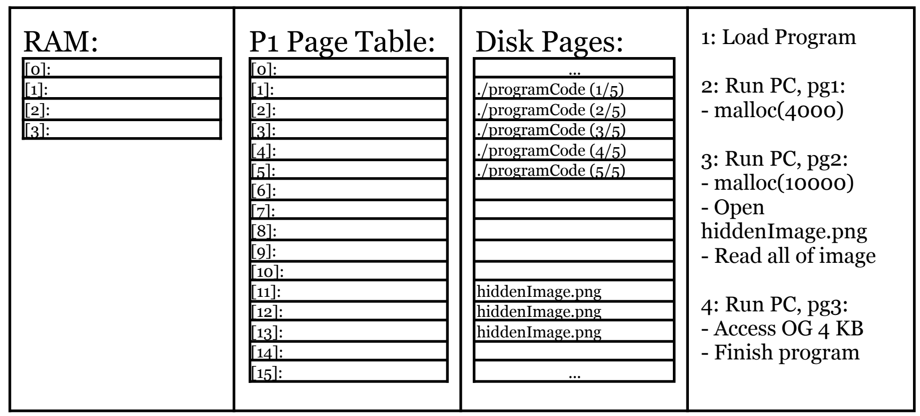
05.c

```
printf("  Start of `array`: %p\n", array);
printf("    End of `array`: %p\n", &(array[(SIZE * SIZE) -1]));
```

2.

3.

Simple Simulation of Page Tables with Disk Pages



Q1: What is the range of possible file sizes for **`hiddenImage.png`**?

Q2: What is the range of possible file sizes for **`./programCode`**?

Q3: What is the size of the heap immediately before the program finishes?