1. The following indicates a part of memory, available for allocation. The memory is divided into segments of fixed sizes of the following sizes.

10 KB	4 KB	20 KB	18 KB	7 KB	9 KB	12 KB	15 KB

- 3 processes to be allocated successively.

A - 12 KB, B - 10 KB, C - 9 KB.

(a) First Fit

10 KB	4 KB	20 KB	18 KB	7 KB	9 KB	12 KB	15 KB
B		A	C				

(b) Best Fit

10 KB	4 KB	20 KB	18 KB	7 KB	9 KB	12 KB	15 KB
B					C	A	

(c) Worst Fit

10 KB	4 KB	20 KB	18 KB	7 KB	9 KB	12 KB	15 KB
		A	B				C

(d) Next Fit

10 KB	4 KB	20 KB	18 KB	7 KB	9 KB	12 KB	15 KB
		A	B		C		

- The algorithm that makes the best use of the memory space is the best fit algorithm; It produces the smallest fragments which still leaves free blocks for large programs.

---

2. A computer with 16 bit address has virtual address space of 64 KB and physical memory of 32 KB. The size of a page is 4 KB.

(a) How many virtual pages and page frames are generated?

i) # of pages = $\frac{\text{total logical size}}{\text{page size}} = \frac{64^{KB} \times 1,024}{4\ KB} = 16,384$ KB

$\simeq$ 16 pages

ii) # of frames = $\frac{\text{total physical size}}{\text{page size}} = \frac{32KB}{4KB} \simeq$ 8 frames

(b) Determine the size of a page table for a computer with 32 bit address, a page size of 4 KB and each entry in the page table requires 4 bytes.

- virtual address space = 32 bit = 2^{32}
- page size = 4 KB = 2^{12}

↳ Page table may consist of 1 million entries

↳ $\frac{2^{32}}{2^{12}} = 2^{20} = 1,048,576 = 1$ MB

↳ If each entry in the table requires 4 bytes

↳ $2^{20} \cdot 4 = \boxed{4\ MB}$

∴ Therefore the size of the table is 4 MB.

---

3. Suppose that we have free segments with sizes:
   6, 17, 25, 14, and 19.

- Place a program with size 13 KB in the free segment using first-fit, best-fit, and worst-fit.
- Indicate which segment it will be in for each of the 3 algorithms.

6 KB	17 KB	25 KB	14 KB	19 KB
	First-Fit	Worst-Fit	Best-fit	

---

4. Consider a logical address space of 8 pages of 1024 words each, mapped onto a physical memory of 32 frames.

(a) How many bits are there in the logical address?

- Logical address space needs 3 bits to represent 8 pages.
  ↳ $2^3 = 8$

- If each page contains 1024 words then offset can be represented by 10 bits
  ↳ $2^{10} = 1024$

↳ Total number of bits to represent logical address is

$$3+10 = \boxed{13 \text{ bits}}$$

(b) How many bits are there in physical address?

↳ If physical memory = 32 frames = 5 bits since $2^5 = 32$.
↳ The offset of 10 bits remains the same.

$\therefore$ $5+10 = \boxed{15 \text{ bits in the physical address}}$

---

5. Suppose the page table for a process currently executing on the processor looks like the following. All addresses are memory byte addresses, and addresses in the main memory and processes start from zero. The page size is 512 bytes.

Virtual Page Number	Valid Bit	Page Frame Number
0	1	4
1	1	7
2	0	1
3	1	2
4	0	8
5	1	0

- What physical address, if any, would each of the following virtual addresses correspond to?

(i) 152

- Integer division of $\frac{152}{\text{page size}} = \frac{152}{512} = 0$

↳ ∴ The page number is 0, offset is 152
↳ table stores page frame number = 4

- Physical address = page frame number * page size + offset
= (4 * 512) + 152
= 2200

(ii) 1121

- Integer division of $\frac{1121}{512} = 2$ = page number

↳ 1121 mod 512 = 97 = offset.
↳ from table
↳ page number 2 has page frame number 1.

- Physical address = (1 * 512) + 97
= 609

(iii) 2499

- Integer division of $\frac{2499}{512} = 4$ = virtual page number.

↳ 2499 mod 512 = 451 = offset
↳ from table
↳ page number 4 has page frame number 8

- Physical address = (8 · 512) + 451

= 4547

---

6. Consider the following segment table:

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

- All addresses are memory byte addresses, and addresses in the main memory and processes start from zero. What are the physical addresses for the following logical addresses (segment number, offset)

(a) 0, 430

↳ Segment: 0 → Base: 219 + 430 (under limit (600)) = 619

(b) 1, 10

↳ Segment: 1 → Base: 2300 + 10 (under limit (14)) = 2310

(c) 2, 500

↳ Segment: 2 → Base: 90 + 500 (over limit)

↳ thus there would be a segmentation fault.

(d) 3, 400

Segment → 3 : Base : 1327 + 400 = 1727

(e) 4, 112

Segment → 4 : Base : 1952 + 112 over limit

↳ thus there would be a segmentation fault

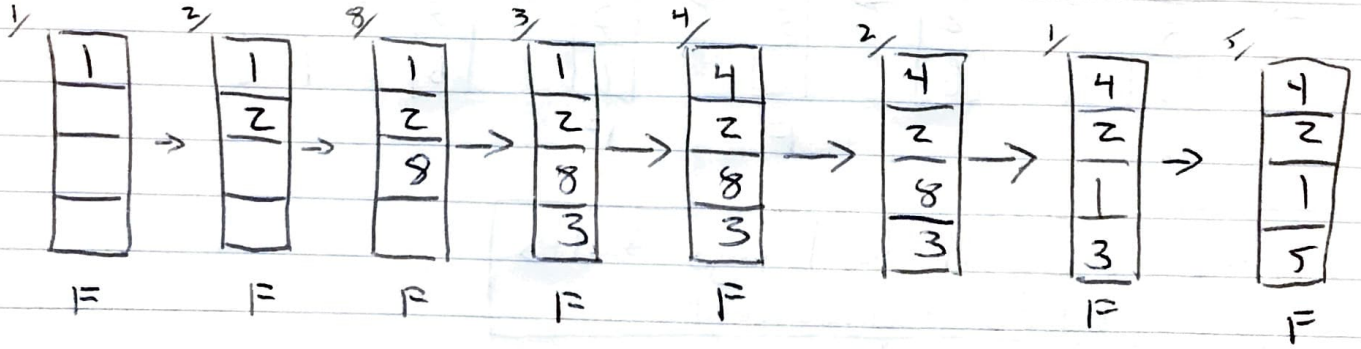
---

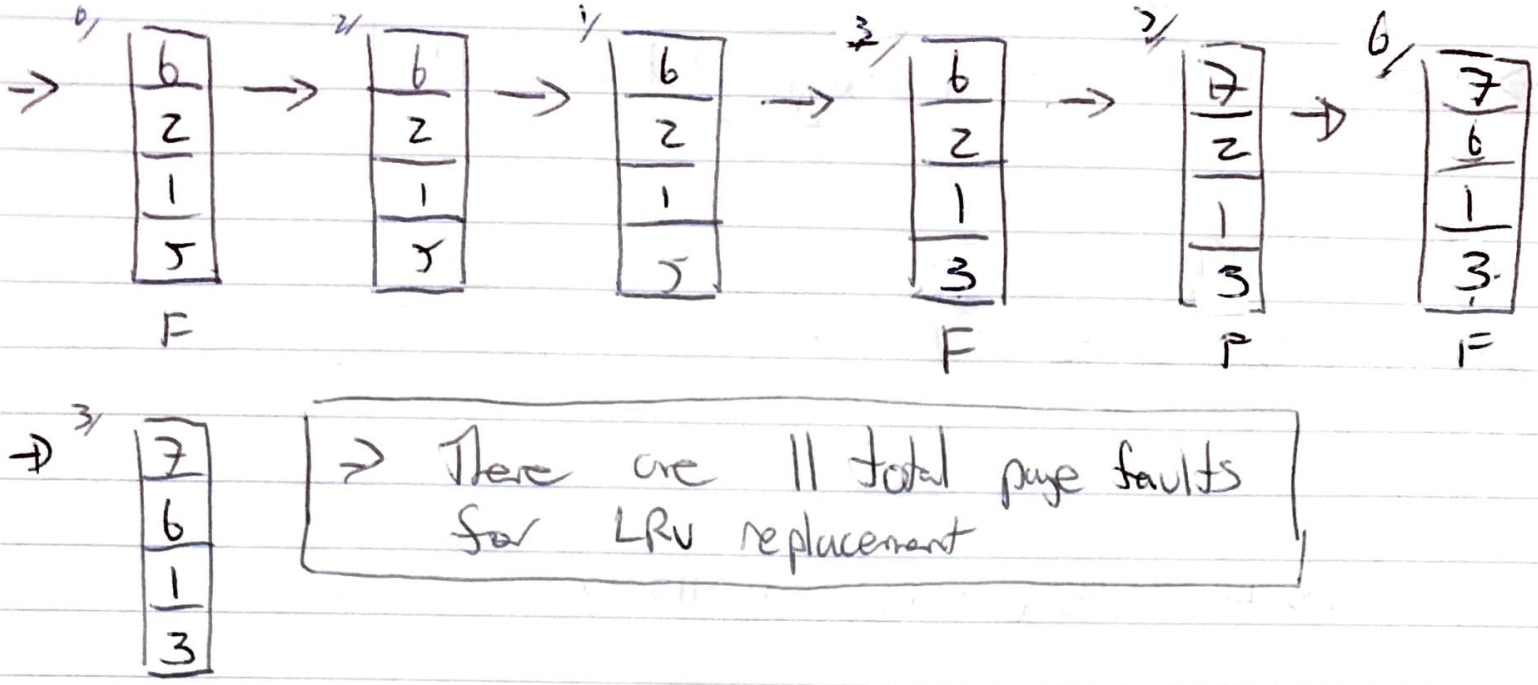
7. Consider the following page-reference string in virtual memory management

: 1, 2, 8, 3, 4, 2, 1, 5, 6, 2, 1, 3, 7, 6, 3

• Assuming four frames, how many page faults would occur for the following replacement algorithms?

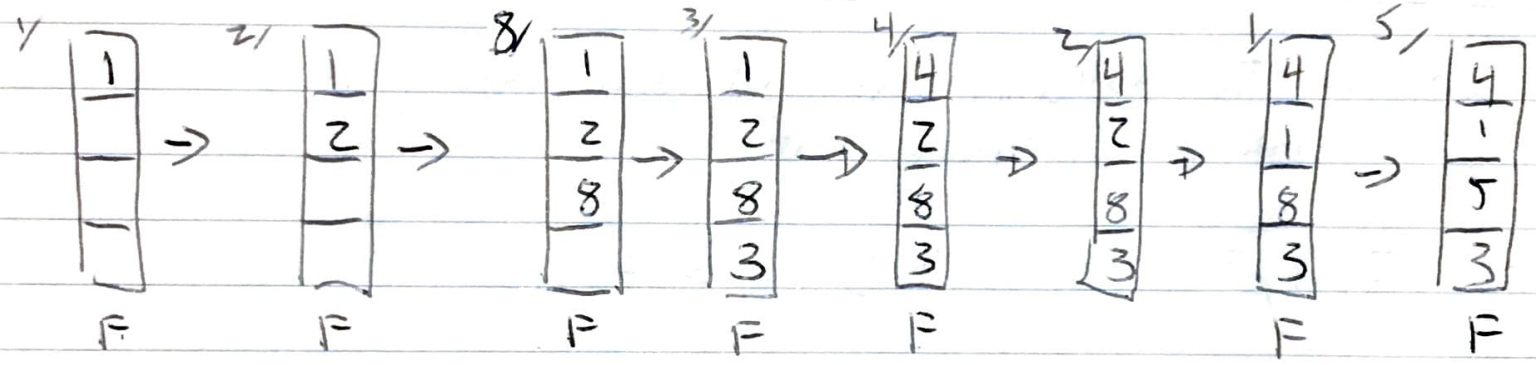
(a) LRU replacement

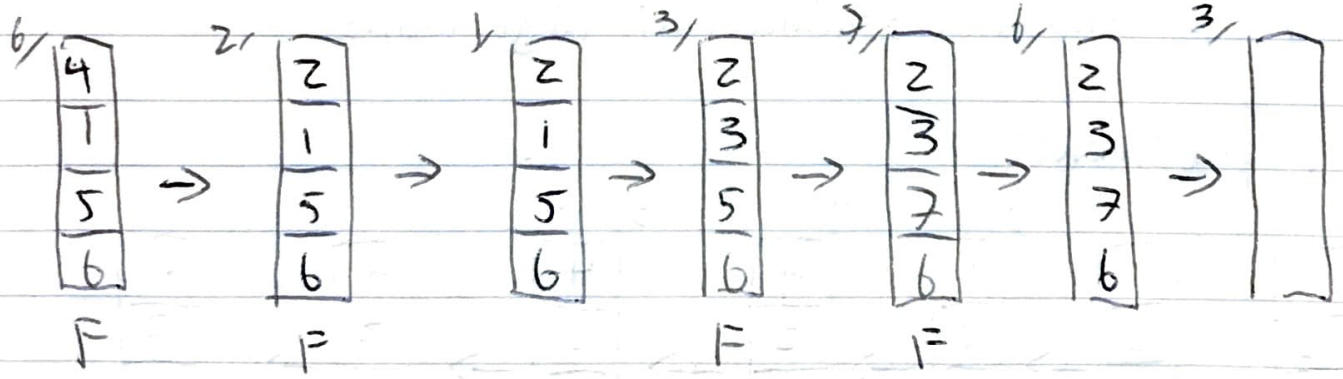




→ There are 11 total page faults for LRU replacement

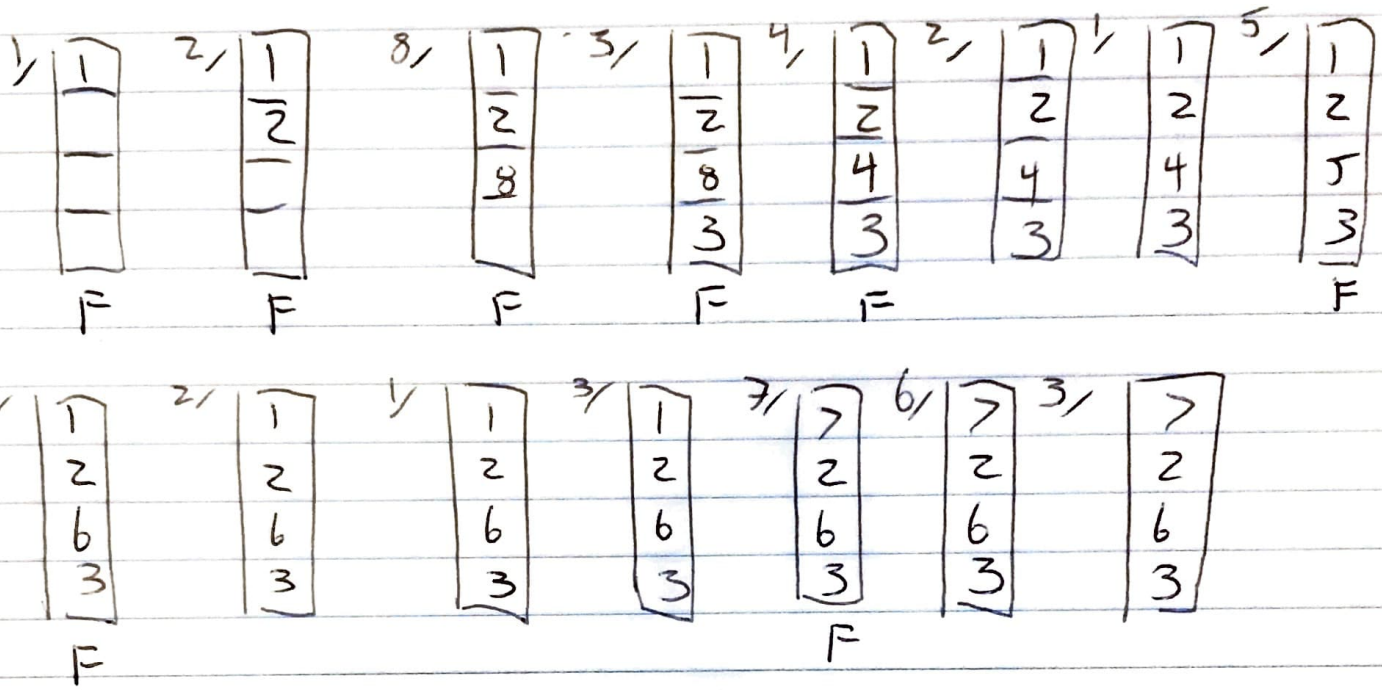
(b) FIFO replacement





→ There are 11 total page faults for FIFO replacement

(c) Optimal Replacement



- There are 8 total page faults for optimal replacement.