Artinkhor Dmitrii

N71

a) Show how the list of free blocks changes after each allocation using the best-fit algorithm.

| 1 | Red | quest | 1 | 16 KiB | ١ | 8 Ki | В | 6 | KiB | 1 | 21 KiB | I | 14 KiB | I | 10 KiB | 1 |
|----|-------|-------|-----|--------|-----|------|---|--------|-----|-----|--------|-----|--------|---|--------|---|
| l' | 9 | KiB | -+- | 16 | -+- | 8 | , | + I | 6 | -+- | 21 | ·+· |) 4 | 1 | 1 | i |
| i | 13.77 | KiB | | 16 | i | 9 | | i | 6 | i | 71 | i | 2 | i | 1 | i |
| I | | KiB | 1 | 16 | I | 1 | | I | 6 | 1 | 7 (| ١ | 3 | I | 1 | I |
| I | 16 | KiB | 1 | 5 | 1 | 1 | | | 6 | I | 21 | I | 3 | I | 1 | I |

b) Show how the list of free blocks changes after each allocation using the worst-fit algorithm.

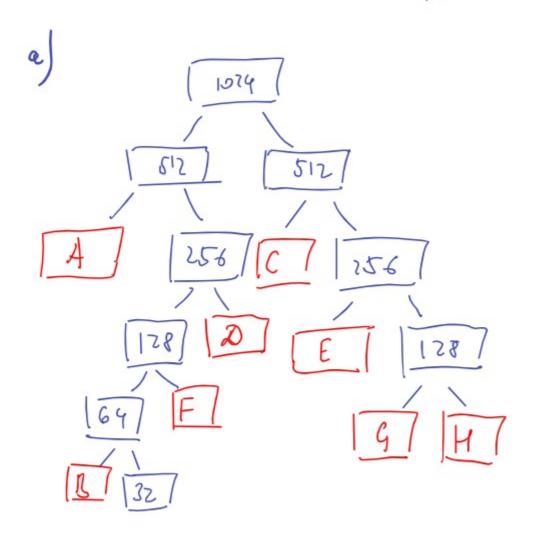
| I | Red | quest | - | 16 Ki | В | 8 KiB | 1 6 | KiB | 21 | KiB | I | 14 KiB | I | 10 KiB | ١ |
|---|-----|-------|-----|-------|----|-------|-------|------|------|-----|----|--------|----|--------|---|
| | 9 | KiB | -+- | 16 | +· | 8 | -+ | 6 | -+ | 12 | +- | 14 | +- | 10 | 1 |
| i | | KiB | i | 5 | i | 8 | i | 6 | i | 25 | i | 14 | i | 10 | i |
| I | 7 | KiB | 1 | 5 | 1 | 8 | 1. | 6 | ١. | 12 | 1 | 7 | 1 | 10 | I |
| I | 16 | KiB | 1 | | - | C | o hit | ello | date | | I | | 1 | | ١ |

c) Show how the list of free blocks changes after each allocation using the first-fit algorithm.

| ļ | Re | quest | 1 | 16 KiB | 8 | 8 KiB | ١ | 6 KiE | 1 | 21 KiB | ١ | 14 KiB | I | 10 KiB | ı |
|---|----|-------|---|--------|---|-------|---|-------|---|--------|----|--------|--------|--------|---|
| ľ | 9 | KiB | 1 | 7 | 1 | 8 | | 6 | 1 | ٦1 | -+ | 14 | +- | 10 | ı |
| ١ | 11 | KiB | 1 | 2 | 1 | 8 | I | 6 | 1 | 10 | ١ | 16 | ١ | 10 | ı |
| 1 | 7 | KiB | 1 | ó | 1 | 6 | I | 6 | I | 10 | 1 | 14 | ١ | 10 | I |
| ١ | 16 | KiB | - | | 1 | | 1 | soft | 1 | locate | 1 | | ١ | | I |

d) Show how the list of free blocks changes after each allocation using the next-fit algorithm.

| - | Request | 1 | 16 KiB | 8 Ki | В | 6 KiB | 1 | 21 KiB | | 14 KiB | 10 Ki | 3 |
|---|---------|---|--------|------|-----|--------|-----|--------|---|--------|-------|---|
| i | 9 KiB | Ī | 7- | P | 1 | G | | 21 | ı | 14 | 1 (0 | |
| ١ | 11 KiB | ١ | 7 | 8 | 1 | 6 | I | 10 | I | 19 | 110 | I |
| 1 | 7 KiB | 1 | 7 | 1 8 | 1 | 6 | 1 | (0) | 1 | 7 | 1 10 | 1 |
| 1 | 16 KiB | 1 | | I | Cal | it all | 970 | te | 1 | - | 1 | 1 |



| | (02 | 9 | |
|---|--------------|---|---------|
| A | 256 | 5 | 12 |
| A | B(32)64) 125 | 5 | 12 |
| A | B(32)64) 125 | C | 522 |
| A | B(35)64) D) | C | 522 |
| A | B(35)64) D) | C | Eliza |
| Ą | B(33) F) 3 | C | Eliza |
| Ą | (B(32) F) 3 | C | E 19/64 |
| A | 1B(32) F) 3 | C | E (G(H |

6) Overall interval fragmentation: 32 KiB Largest church to allocate: 32 KiB

c) After deallocation of F me get 2 charks of 32 and 69 kib, but we can't merge them because they have different sizes. So we cannot fulfil the reguest for 80 kib.

17.3

 a) Show how the pages are mapped to two frames using the First-In-First-Out (FIFO) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|----|---|---|---|---|---|---|---|---|
| frame 0 | 3 | 3, | 2 | 2 | (| 1 | 2 | 2 | 1 | |
| frame 1 | | 1 | T | 4 | 4 | 3 | 3 | 4 | 4 | 3 |

Show how the pages are mapped to three frames using the First-In-First-Out (FIFO) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|---|---|---|-----|-----|-----|---|---|---|
| frame 0 | 3 | 3 | 3 | 4 | 9 | 4 | 4 | 4 | 4 | 4 |
| frame 1 | | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 |
| frame 2 | | | 2 | 2 | - 2 | . 2 | - 2 | 2 | 1 | 1 |

b) Show how the pages are mapped to two frames using Belady's Optimal (BO) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|---|---|---|---|---|---|---|---|---|
| frame 0 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 4 | 1 | 1 |
| frame 1 | | (| 1 | 4 | 1 | 3 | 3 | 2 |) | 3 |

Show how the pages are mapped to three frames using Belady's Optimal (BO) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|---|---|---|---|---|---|---|---|---|
| frame 0 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 |
| frame 1 | | 1 | 1 | 1 | 1 | 1 | - | 1 | 1 | P |
| frame 2 | | | 2 | 4 | 4 | 9 | 4 | 4 | 4 | 4 |

c) Show how the pages are mapped to two frames using the Least Recently Used (LRU) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|---|---|---|---|---|---|---|---|---|
| frame 0 | 3 | 2 | 1 | 2 | (| 1 | 2 | S | 1 | |
| frame 1 | | 7 | 1 | 4 | 4 | 3 | 3 | 4 | 4 | 3 |

Show how the pages are mapped to three frames using the Least Recently Used (LRU) page replacement algorithm.

| reference string | 3 | 1 | 2 | 4 | 1 | 3 | 2 | 4 | 1 | 3 |
|------------------|---|---|-----|---|---|---|---|---|---|---|
| frame 0 | 3 | 3 | 3 | 4 | 4 | 4 | 2 | 2 | 2 | 3 |
| frame 1 | | 1 | - 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 |
| frame 2 | | | 2 | 2 | 2 | 3 | 3 | 3 | 1 | 1 |