Problem 1 (20 points)

The virtual address of a virtual memory system with both paging and segmentation is expressed as \((s_1, s_2, p, d)\) (see Figure 1) with \(|s_1|, |s_2|, |p|, |d|\) the length in bits of the four components of the address. The system has the following characteristics:

a. Word size is 32 bits.
b. An entry in the page or segment table is one word.
c. Page size is 512 words.
d. The page table is at most one page.

b. An entry in the page or segment table is one word.

(Q1) What is the value of \(|d|\)?
(Q2) What is the maximum number of pages per segment and what is the corresponding value of \(|p|\)?
(Q3) Using the values of \(|d|\) and \(|p|\) determined above which of the following choices for \(|s_1|\) and \(|s_2|\) is preferable. Explain.

(a) \(|s_1| = d\) and \(|s_2| = 32 - |s_2| - |p| - |d|.
(b) \(|s_2| = d\) and \(|s_1| = 32 - |s_1| - |p| - |d|.

Problem 2 (20 points)

Assuming a physical memory of of four page frames, give the number of page faults for the reference string \(abgadeabadegde\) for each of the following policies assuming that initially all page frames are empty.

a. MIN (Ideal/Optimal Replacing Policy)
b. FIFO
c. Clock replacement
d. Second-chance (assume that all accesses to page b are are write requests).
e. LRU
f. WS with \(\Delta = 3\)

Problem 3 (20 points)

Consider the reference string \(abcedebedcbdddbddd\). Assuming the working set replacement policy determine the minimum window size \(\Delta\) such that the string generates at most five page faults. Show which pages are in memory at each reference. Mark page faults with an asterisk.
Figure 1: Virtual Address
Problem 4 (40 points)

Devise data structure and algorithms to:
1. Maintain the working set of a process
2. Select a replacement page using the working set principles.

Problem 5 (20 points)

A two-dimensional $512 \times 512$ matrix is stored in row order in a paged virtual memory system with a page size of 512 words and 16 frames of physical memory. Assuming FIFO page replacement, how many page faults will be generated when the entire matrix is processed (i) by row (ii) by column. Would the results differ under LRU policy?

Problem 6 (30 points)

Consider static sharing in a purely pages system (no segmentation) under the following conditions: a process $P_a$ requests programs $Q_1$ and $Q_2$, temporary storage $T_1$, and data $D_1$ of size of $q_1, q_2, t_1$ and $d_1$ pages, respectively. These are combined and linked together into an object module of size $q_1 + q_2 + t_1 + d_1$ pages in that order and then loaded into memory. While $P_a$ is still active, another process $P_b$ request loading programs $Q_1'$ and $Q_2'$, temporary storage $T_1'$, and data $D_1$ of size of $q_1', q_2', t_1'$ and $d_1'$ pages, respectively, with $q_1' > q_1$ and $t_1' < t_1$. $Q_2$ and $D$ are shared by $P_a$ and $P_b$.

1. Describe possible contents of the page tables for $P_a$ and $P_b$.
2. What problem is created when a new process $P_c$ is loaded and wants to share $Q_1'$ and $T_1'$?