Part 1: (33 points - 3 points for each problem)

(C) 1. Which strategy is used in the Banker’s algorithm for dealing with deadlocks?
   (A) Deadlock Ignorance (B) Deadlock Detection (C) Deadlock Avoidance (D) Deadlock Prevention

(D) 2. Which deadlock condition does ordering resources numerically attack?
   (A) Mutual exclusion (B) Hold and wait (C) No preemption (D) None of the above

(B) 3. Which of the memory allocation schemes are not subject to internal fragmentation?
   (A) Multiple Contiguous Fixed Partitions (B) Segmentation (C) Paging (D) None of above

(D) 4. A system has 256 MB memory. The time to read or write a 32-bit memory word is 10 nsec.
   Assume the total processes take 3 times of memory taken by holes. What is the time needed to
   eliminate holes by compaction?
   (A) 167.772 ms (B) 335.544 ms (C) 503.316 ms (D) None of the above

(A) 5. A computer with physical address space of $2^{24}$ words has the frame size of 4096 ($2^{12}$) words.
   If the hexadecimal physical address is 123456, the frame number in hexadecimal would be:
   (A) 123 (B) 456 (C) 1234 (D) 123456

(D) 6. If there are 128K pages and the page size is 32K words, the length of logical address is:
   (A) 24 bits (B) 28 bits (C) 30 bits (D) 32 bits

(C) 7. A system has 64 virtual pages mapping into 16 physical frames in the following equation:
   frame number = page number % 16. Each page has 1K words. If the virtual address is 1010101001111101,
   the physical address would be:
   (A) 11001000111101 (B) 10101000111111 (C) 10101000111101 (D) None of the above

(A) 8. A machine with 512 ($2^9$) MB memory has a 32-bit memory word. The frame size is 8 K ($2^{13}$)
   words. How many bits are used to indicate the frame number?
   (A) 14 bits (B) 16 bits (C) 20 bits (D) None of the above

(A) 9. A system with a 32-bit virtual address. Each page size is 16 KB words. Each table entry takes
   4 bytes. What is the size of the page table?
   (A) 1 MB (B) 2 MB (C) 4 MB (D) 8 MB

(B) 10. The modified (dirty) bit cannot be used for the purpose of:
   (A) Implementing NRU page replacement algorithm
   (B) Dynamic allocation of memory used by one process to another
   (C) Reducing the average time required to service page faults
   (D) None of the above.

(D) 11. Which statement about segmentation is false?
   (A) There are many linear address spaces.
   (B) The total address space can exceed the size of physical memory.
   (C) Sharing of procedures between users is facilitated.
   (D) None of the above
Part 2: (67 points)

1. (8 pts.) P is a set of processes. R is a set of resources. E is a set of request or assignment edges. The sets P, R, and E are as follows:

\[ P = \{P_1, P_2, P_3\}, \quad R = \{R_1, R_2, R_3\}, \quad E = \{P_1 \rightarrow R_1, P_1 \rightarrow R_2, P_2 \rightarrow R_2, P_2 \rightarrow R_3, P_3 \rightarrow R_2, P_3 \rightarrow R_3, R_1 \rightarrow P_2, R_2 \rightarrow P_2, R_3 \rightarrow P_1\}. \]

R_1 has one instance. R_2 has two instances. R_3 has one instance.

(a) Draw the resource-allocation graph.

(b) Is there any deadlock in this situation? Briefly Explain.

Ans:

(a) The resource-allocation graph is shown as follows:

(b) Consider the resource-allocation graph. There are four cycles in the system:

- P_1 \rightarrow R_1 \rightarrow P_2 \rightarrow R_3 \rightarrow P_1, P_1 \rightarrow R_2 \rightarrow P_2 \rightarrow R_3 \rightarrow P_1, P_2 \rightarrow R_2 \rightarrow P_2,
- P_2 \rightarrow R_3 \rightarrow P_1 \rightarrow R_1 \rightarrow P_2.

P_1 cannot finish because P_1 needs R_1 and R_2 but can only acquire R_2 while R_1 is held by P_2. P_2 cannot finish because P_2 needs R_2 and R_3 but can only acquire R_2 while R_3 is held by P_1. P_3 cannot finish because P_3 needs R_1 and R_3 but can only acquire R_2 while R_3 is held by P_1. P_1, P_2, and P_3 cannot progress. The deadlock occurs.

2. (a) (2 pts.) What is Belady’s anomaly?

(b) (2 pts.) Give two page-replacement algorithms that do not suffer from Belady’s anomaly.

(c) (3 pts.) What is the working set model?

(d) (2 pts.) What is the cause of thrashing?

(e) (2 pts.) How does the system detect thrashing?

Ans:

(a) more page frames might not always have fewer page faults. This is called Belady’s anomaly.

(b) LRU and optimal algorithms suffer from Belady’s anomaly.

(c) Paging systems keep each process. working set in memory before letting the process run. This approach is called the working set model.

(d) Thrashing is caused by under-allocation of the minimum number of pages read required by a process, forcing it to continuously page fault.

(e) The system can detect thrashing by evaluating the level of CPU utilization as compared to the level of multiprogramming.
3. (10 pts.) Consider a swapping system in which memory consists of 540K as shown below:

\[
\begin{array}{cccccccc}
\text{xxxx } & \text{P3 } & \text{xxxx } & \text{P5 } & \text{xxxxx } & \text{P7 } & \text{xx } \\
0 & 60 & 140 & 220 & 320 & 420 & 500 & 540
\end{array}
\]

where xx.. indicates the unused memory.

Note that P3, P5, and P7 are processes in memory. Assume that process P3 was just swapped into memory and that new processes arrive in the order P8, P9, P10, P11 and are of size 70K, 50K, and 80K, 40K respectively. How would each of the first-fit, next-fit, best-fit, and worst-fit algorithms place processes of P8, P9, P10, and P11? If a process won’t fit, write 'out of memory' in the appropriate slot.

Ans:

first fit:

\[
\begin{array}{cccccccc}
\text{P9 } & \text{x } & \text{P3 } & \text{P8 } & \text{x } & \text{P5 } & \text{P10 } & \text{x } & \text{P7 } & \text{P11 }\\
0 & 50 & 60 & 140 & 210 & 220 & 320 & 400 & 420 & 500 & 540
\end{array}
\]

next fit:

\[
\begin{array}{cccccccc}
\text{xxx } & \text{P3 } & \text{P8 } & \text{x } & \text{P5 } & \text{P9 } & \text{x } & \text{P7 } & \text{xx }\\
0 & 60 & 140 & 210 & 220 & 320 & 370 & 410 & 420 & 500 & 540
\end{array}
\]

P10: out of memory

best fit:

\[
\begin{array}{cccccccc}
\text{P9 } & \text{x } & \text{P3 } & \text{P8 } & \text{x } & \text{P5 } & \text{P10 } & \text{xx } & \text{P7 } & \text{P11 }\\
0 & 50 & 60 & 140 & 210 & 220 & 320 & 400 & 420 & 500 & 540
\end{array}
\]

worst fit:

\[
\begin{array}{cccccccc}
\text{P11 } & \text{x } & \text{P3 } & \text{P9 } & \text{x } & \text{P5 } & \text{P8 } & \text{xx } & \text{P7 } & \text{xx }\\
0 & 40 & 60 & 140 & 190 & 220 & 320 & 390 & 420 & 500 & 540
\end{array}
\]

P10: out of memory

4. (10 pts.) The time required to read a word in memory (including the time to access associative memory) is 80 nanoseconds when the page number is in the associative memory. The access time of associative memory is 10 nanoseconds.

(a) Find a formula that expresses the effective access time as a function of the hit ratio.

(b) What hit ratio is needed to achieve the effective access time to 101 nanoseconds?

Ans:

(a) Let e and t be the access time to associative memory and ordinary memory respectively. Assume the hit ratio be h.

\[
e + t = 80, \ 10 + t = 80, \ t = 70
\]

Effective Access Time = \( h \times 80 + (1 - h) \times (10 + 2 \times 70) = 80h + (1 - h) \times 150 = 80h + 150 - 150h = 150 - 70h \)

(b) \( 150 - 70h \leq 101, \ 70h \geq 49, \ h \geq 70% \) The hit ratio should be 70%.

5. (8 pts.) A computer has four page frames. The time of loading, time of last access, and the R and M bits for each page are as shown below (the times are in clock ticks):

<table>
<thead>
<tr>
<th>Page</th>
<th>Loaded</th>
<th>Last Referenced</th>
<th>R M</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>140</td>
<td>270</td>
<td>1 1</td>
</tr>
<tr>
<td>1</td>
<td>230</td>
<td>260</td>
<td>0 1</td>
</tr>
<tr>
<td>2</td>
<td>126</td>
<td>280</td>
<td>1 0</td>
</tr>
<tr>
<td>3</td>
<td>162</td>
<td>265</td>
<td>0 0</td>
</tr>
</tbody>
</table>

(a) Which page will NRU replace?

(b) Which page will FIFO replace?

(c) Which page will LRU replace?

(d) Which page will second chance replace?

Ans:

(a) Page 3 because R and M bits are 0.

(b) Page 2 because it is oldest (loaded at 126).

(c) Page 1 because is least recently referenced (referenced at 260).

(d) Page 3 because it is oldest and not referenced (loaded at 162 and R = 0).
6. (8 pts.) A small computer has five page frames. At the first clock tick, the R bits are 11001 (page 2 and 3 are 0). A subsequent clock ticks, the values are 10110, 00101, 00101, 11000, 01011, 10101, and 11001

(a) If the aging algorithm is used with an 8-bit counter, give the values of the counters after the last tick.

(b) Which page would be selected to be removed from memory?

Ans:

(a) 

<table>
<thead>
<tr>
<th>R bits</th>
<th>Page 0</th>
<th>Page 1</th>
<th>Page 2</th>
<th>Page 3</th>
<th>Page 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>11001</td>
<td>10000000</td>
<td>10000000</td>
<td>00000000</td>
<td>00000000</td>
<td>10000000</td>
</tr>
<tr>
<td>10110</td>
<td>11000000</td>
<td>01000000</td>
<td>10000000</td>
<td>10000000</td>
<td>01000000</td>
</tr>
<tr>
<td>00101</td>
<td>01100000</td>
<td>00100000</td>
<td>11000000</td>
<td>01000000</td>
<td>10100000</td>
</tr>
<tr>
<td>00101</td>
<td>00110000</td>
<td>00010000</td>
<td>11100000</td>
<td>00100000</td>
<td>11010000</td>
</tr>
<tr>
<td>11000</td>
<td>10011000</td>
<td>10001000</td>
<td>01110000</td>
<td>00010000</td>
<td>01101000</td>
</tr>
<tr>
<td>01011</td>
<td>01001100</td>
<td>00110000</td>
<td>10011100</td>
<td>10001000</td>
<td>10110100</td>
</tr>
<tr>
<td>10101</td>
<td>11011001</td>
<td>10100110</td>
<td>01001100</td>
<td>00101000</td>
<td>10111010</td>
</tr>
<tr>
<td>11001</td>
<td>11010011</td>
<td>10110001</td>
<td>01001110</td>
<td>00100010</td>
<td>11101101</td>
</tr>
</tbody>
</table>

(b) Page 3 has the smallest value. It will be evicted.

7. (12 pts.) Suppose there are 8 virtual pages and 4 page frames. Determine the number of page faults that will occur with the reference string 0 2 3 1 4 2 5 6 1 3 2 4 7 1 2, if the page frames are initially empty, using each of the following page replacement algorithms: (a) FIFO (b) LRU (c) Optimal.

Ans:

(a) FIFO - 14

(b) LRU - 14

(c) Optimal - 10