STUDENT NAME:
STUDENT ID:

<table>
<thead>
<tr>
<th>Problem 1</th>
<th>Problem 2</th>
<th>Problem 3</th>
<th>Problem 4</th>
<th>Problem 5</th>
<th>Problem 6</th>
<th>Total Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>100</td>
</tr>
</tbody>
</table>
Problem 1: (15 points) Consider a system containing two types of resources. There are three instances of type $R_1$ and two instances of type $R_2$. There are two processes $P_1$ and $P_2$ in the system. The maximum resource requirement of $P_1$ is 1 for $R_1$ and 1 for $R_2$, and that of $P_2$ is 3 for $R_1$ and 2 for $R_2$. Using resource allocation graphs identify all possible deadlock states for this system. (In your answer label all nodes in the graphs appropriately. Show separate graphs if there are multiple instances of deadlock.)
Problem 2: (15 points) Consider a system that has four types of resources, and the number of instances of these in the system are (6, 4, 7, 4).

Consider the following state of the system: Currently available resources are (3, 0, 2, 1).

<table>
<thead>
<tr>
<th>Process</th>
<th>Allocation</th>
<th>Remaining Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0 1 3 0</td>
<td>1 3 2 1</td>
</tr>
<tr>
<td>P2</td>
<td>0 2 0 2</td>
<td>4 0 0 0</td>
</tr>
<tr>
<td>P3</td>
<td>1 0 2 0</td>
<td>1 2 5 1</td>
</tr>
<tr>
<td>P4</td>
<td>2 1 0 1</td>
<td>2 0 1 0</td>
</tr>
</tbody>
</table>

(a) (10 points) Is this system in a safe state? Justify your answer. If the system state is safe, give a completion sequence for the processes.

(b) (10 points) If a request from process P3 arrives for (0, 0, 1, 0), can the request be immediately granted? Justify your answer.
Continue answer for Problem 2...
Problem 3: (20 points)
Part A: (3 Points) Identify at least three items of information maintained in a page-table entry.

Part B: (4 points) Page locking prevents a memory page from being replaced on page-faults. Give two examples of cases which require page locking and briefly explain why.

Part C: (5 points) Indicate if the following statements are True or False.

- (1 point) A smaller page-size leads to a smaller size for page-table. True False
- (1 point) A smaller page-size leads to more TLB misses. True False
- (1 point) Considering that 4GB physical memories are common and cheap, is virtual memory really needed for machines with 32-bit addresses? True False
- (1 point) A TLB miss can occur even though the requested page is in memory. True False
- (1 point) Kernel maintains only one page-table for all processes in the system. True False
Part D: (8 points) A computer uses a two-level page table and virtual addressing. Virtual addresses are split into a 10-bit top-level page table field, a 12-bit second level page table field, and a 10-bit offset.

(i). (2 points) How large is the virtual address space?

(ii). (2 points) How large are the pages and how many are there in the address space? Provide your answers in powers of 2.

(iii). (4 points) Assume that a page-table entry requires 4 bytes. How many memory pages are required for the first-level page table? How many pages are required for a second-level page table?
Problem 4: (15 points) Suppose that a computer system has primary memory with access time of 200 nanoseconds. It also supports demand paged virtual memory. Assume the following:

(1) It takes 20 nanoseconds to search the TLB.
(2) The average time to read or write a page on disk is 10 milliseconds.

Answer the following questions: (Just write the formula, no need to use a calculator. You must indicate the basis for each cost component in your formula, as it was done in the solutions for your homework problems.)

a (2 points) What is the memory access time when we have TLB hit?

b (4 points) What is the memory access time when we have a TLB miss and the page-table contains a valid mapping for the accessed page, i.e. no page-fault is encountered.
c (4 points) What is the memory access time when we have a TLB miss, there is a page-fault, and the page selected for replacement is not dirty.

d (5 points) What is the memory access time when we have a TLB miss and also a page-fault, and the page selected for replacement is dirty.
Problem 5 (20 points) : Consider the following program:

```pascal
var A, B: array[1..1000] of integer;
for i := 1 to 1000 do
    B[i] = A[i];
endfor;
for i := 1000 to 1 do
    A[i] = B[i];
endfor;
```

An integer occupies a word and page size is 200 words. Assume that the code and the variable \( i \) are placed in logical page 0, which is always kept in memory so that access to \( i \) and instruction fetch do not produce a page fault. Assume that array A is stored in logical pages 1 through 5, and B is stored in pages 6 through 10. The main memory is initially empty.

(a) What is the reference string generated by this program? (Ignore access to page 0 for instructions and access to \( i \).)

(b) When 4 frames are given to this program (in addition to the one for logical page 0 containing code and \( i \)) and under LRU replacement policy determine the following:

- Total number of page faults.
- State of the 4 page frames after handling each page fault.
Continue answer for Problem 5...
Problem 6: (15 points) Consider a system using the WSclock scheme for working set based virtual memory management. Suppose that the current virtual time 2000. This system had 8 page frames. Figure 1 shows the position of the clock hand, and the values of the last-time-of-use and the $R$ and $M$ bits for each frame.

![Diagram showing frame status at virtual time 2000](image)

Figure 1: Frame status at the time of a page fault at virtual time = 2000

Suppose that instead of the clock interrupt a page-fault occurs at virtual time 2000. For $\tau$ equal to 200:

1. (4 points) Indicate the frame whose page is replaced.
2. (8 points) Show the new contents of the frame-list.
3. (3 points) Determine which of the pages will be candidates for removal from the working set.
Continue answer for Problem 6...