<table>
<thead>
<tr>
<th>Problem #</th>
<th>Points</th>
<th>Your Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>120</strong></td>
<td></td>
</tr>
</tbody>
</table>

Name_______________________________________

Student ID: _________________________________

Section: __________
1. Recursion:
(5 points)
Write a recursive function, `power`, that calculates the power, \( pow \) of \( n^{pow} \).

Answer:
```c
int power(int n, int pow)
{
    if (pow == 0)      /* pow(x, 0) returns 1 */
        return(1);
    else
        return(power(n, pow - 1) * n);
}
```

2. Dijkstra’s algorithm:
(10 points)
Using Dijkstra’s algorithm answer the following questions:

![A network diagram showing cities and distances between them.]

(a) Which node(s) are involved in the first iteration of finding the shortest path from Melbourne to all other cities using Dijkstra’s algorithm?

(b) Which node(s) are involved in the second iteration of finding the shortest path from Melbourne to all other cities Dijkstra’s algorithm?

(c) Which node is the last to find the shortest path from Melbourne?
Answer:
   a) Adelaide, Sydney
   b) Darwin
   c) Darwin

3. Merge sort:
Given the following array: 4 29 44 22 37 94 38 84
(10 points)
   a) Using a merge sort, what is the result of the first iteration of the merge sort?

   b) What is the result of the second iteration of the merge sort?

   c) Is there a more efficient sorting algorithm and if so, why is it more efficient?

Answer:
   a) 4 29 44 22, 37 94, 38 84
   b) 4 29, 44 22, 37 94, 38 84
   c) Quicksort algorithm is more efficient because it takes the first element, a_i, in the array and partitions the initial array into a sublist whose elements are less than a_i and another sublist whose elements are more than a_i. This results in two partially sorted subarrays after the first iteration which is not the case in merge sort.

4.
Let A={1,2,3} and B={1,2,3,4}. The Relations R={((1,1), (2,2), (3,3))} and R2={((1,1), (1,2), (1,3), (1,4))}.
Find
   a) R1 ∪ R2
   b) R1 ∩ R2
   c) R1 – R2
   d) R2 – R1

Solution:
   R1 ∪ R2 = {((1,1), (1,2), (1,3), (1,4), (2,2), (3,3))},
   R1 ∩ R2 = {((1,1))},
   R1 – R2 = {((2,2), (3,3))},
   R2 – R1 = {((1,2), (1,3), (1,4))},
5. Find a rule that can produce the terms of a sequence if the first 10 terms are: 5, 11, 17, 23, 29, 35, 41, 47, 53, 59?

Solution:
Nth term = 6n – 1.

6. Find: AB

Solution:

\[
AB = \begin{bmatrix}
14 & 4 \\
8 & 9 \\
7 & 13 \\
8 & 2
\end{bmatrix}
\]

7. Write using pseudo code how to search for an element in an array.

Answer:
Procedure linear search(x: integer, a1, a2, … an: distinct integers)
i := 1
while (i <= n and x != ai)
i := i + 1
if i <= n then location := i
else location := 0
{location is the subscript of term that equals x, or is 0 if x is not found}

8. Write using pseudo code how to find the maximum element in an array.

Answer:
Procedure max(a1, a2, … an: integers)
Max := a1;
For i: = 2 to n
If max < ai then max := ai

9. Write using pseudo code how to search for an element in a sorted array using binary search.

Answer:
Procedure binary search(x: integer, a1,a2,…an: increasing integers)
i:=1
j:=n
while i<j
begin
m:=(i+j)/2
if x>am then i:=m+1
else j:=m
end
if x=ai then location :=i
else location :=0

9. Draw out on paper how the binary tree evolves for the input sequence:

```
5 7 3 6 9 2
```

Solution:

```
      5
     / 
   3   7
  / 
2   6
```

10. Given the graph give the following traversals

```
      +  *  / 
     /   -   -
   +   +   +  -
A   B   C   D   E   F   G   H
```
a) Preorder  
b) In order  
c) Post order  

Solution:
a) + * - A B + C D / + E F - G H  
b) A - B * C + D + E + F / G - H  
c) A B - C D + * E F + G H - / +

11. Use an adjacency Matrix to represent the graph

Solution

\[
\begin{bmatrix}
0 & 1 & 1 & 1 \\
1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
\end{bmatrix}
\]

12. Explain if these graphs do or do not exhibit Hamilton Circuits and why

a)
Solution

a) There is no Hamilton circuit because there is a vertex of degree (e).

b) The degree of all the vertices is two, so every edge incident with these vertices must be part of any Hamilton circuit. No Hamilton circuit can exist in H, because any Hamilton circuit would have to contain four edges incident with c, which is impossible.