Fun Question: (1 bonus point) Enrico Fermi

Question 1: (20 points)
(a-f) F T T F F F (g) $\Theta(n), \Theta(n \lg n), \Theta(n^2), \Theta(2^n)$    
(h) search function    (i) many

Question 2: (15 points)
(a) (6 points)

<table>
<thead>
<tr>
<th></th>
<th>4</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-Count(T, 1, P, 4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U-Count(T, 3, P, 4)</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>U-Count(T, 8, P, 3)</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

(b) (3 points)
Counts the number of mismatches between P[1..m] and T[k,k+m-1], component-wise.

(c) (2 points)
Dominant Operation: __ key comparison (P[j] = T[k+j-1])  [line 4] _____

(d) (4 points)
Time Complexity: __$\Theta(m)$_______

Question 3: (15 points)
(a) [5]  Search Tree  (DIY)
(b) [4]  Eve: 1  Cathy: 3  Bunny: 3 (unsuccessful)  John: 4
(c) [3]  $(1 + 2 + 2 + 3 + 3 + 3 + 3 + 4 + 4) / 9 = 25/9 = 2.78$
(d) [3]  $(3 + 3 + 3 + 3 + 3 + 4 + 4 + 4 + 4) / 10 = 34/10 = 3.4$
Question 4: (10 points)

Given a database with the following 3 tables: \( \{SI, CI, EN\} \). You should use these short table names to save space and writing. (Use the reverse blank pages, if necessary)

<table>
<thead>
<tr>
<th>SI (STUDENT-INFO)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student-ID</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI (COURSE-INFO)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course-ID</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EN (ENROLMENT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student-ID</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

(Note: In my answers, I use S-ID and C-ID for Student-ID and Course-ID.)

(a) [4 pts] List the Course-ID, Day, Hour of all courses taught in the venue “USP-SR1”.

SQL Query: 
```
SELECT CI.C-ID, CI.Day, CI.Hour
FROM CI
WHERE (CI.Venue = "USP-SR1");
```

Using Basic Primitives:
```
A1 ← e-select from CI where (Venue = "USP-SR1");
Ans ← e-project C-ID, Day, Hour from A1;
```

(b) [6 pts] List the Student-ID, Major of FASS students who have classes in “USP-SR1”.

SQL Query: 
```
SELECT SI.S-ID, SI.Major
FROM SI, CI, EN
WHERE (SI.Faculty = "FASS") AND
(CI.Venue = "USP-SR1") AND
(SI.S-ID = EN.S-ID) AND
(CI.C-ID = EN.C-ID);
```

Using Basic Primitives:
```
B1 ← e-select from CI where (Venue = "USP-SR1");
B2 ← e-select from SI where (Faculty = "FASS");
B3 ← e-join B1 and EN where (B1.C-ID = EN.C-ID);
B4 ← e-join B2 and B3 where (B2.S-ID = B3.S-ID);
Ans ← e-project SI.S-ID, SI.Major from B4;
```

~~~ END OF QUIZ ~~~