Exam 1 Solution

Wednesday, October 2, 2002.

Problem 1 - 8 are 5 points each.

1. Secondary memory is volatile. (b) False.
2. A file is a sequence of bytes. (a) True.
3. A nibble is 6 bits. (b) False.
4. An algorithm can have an infinite loop. (b) False.
5. Is the following a valid algorithm for washing hair instruction?
   1. Lather.
   2. Rinse.
   3. Repeat.

   (b) False.

6. Which of the following is often described as the “brain” of the computer? (c) CPU
7. If the memory has 32 words, how many bits does an address bus need to carry at a time? (c) \( \log_2 32 = 5 \) bits.
8. How many times does the following loops display Hi.?

   \[
   \begin{align*}
   x &= 6 \\
   \text{While } & (x > 0) \\
   & \text{Display Hi.} \\
   & \text{Decrement } x \text{ by 2} \\
   & \text{End_WWhile}
   \end{align*}
   \]

   (c) 3 times.
9. (20 points) Show how you derived your answers. Answer four out five questions below:

(a) Convert the decimal number 66 to binary number.

\[
\begin{align*}
66 \div 2 &= 33 \quad \text{Remainder 0} \quad \text{bit 0} \\
33 \div 2 &= 16 \quad \text{Remainder 1} \quad \text{bit 1} \\
16 \div 2 &= 8 \quad \text{Remainder 0} \quad \text{bit 2} \\
8 \div 2 &= 4 \quad \text{Remainder 0} \quad \text{bit 3} \\
4 \div 2 &= 2 \quad \text{Remainder 0} \quad \text{bit 4} \\
2 \div 2 &= 1 \quad \text{Remainder 0} \quad \text{bit 5} \\
1 \div 2 &= 0 \quad \text{Remainder 1} \quad \text{bit 6}
\end{align*}
\]

Write the answer from bit 0 to bit 6 or right to left. So the answer is \(66_{10} = 1000010_2\).

(b) Convert the binary number 10101 to decimal number.

\[
\begin{align*}
1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\
= 1 \times 16 + 0 + 1 \\
= 17
\end{align*}
\]

The answer is \(10101_2 = 17_{10}\).

(c) Convert the decimal number 77 to hexadecimal number.

\[
\begin{align*}
77 \div 16 &= 4 \quad \text{Remainder 13}_{10} = D_{16} \quad \text{bit 0} \\
4 \div 16 &= 0 \quad \text{Remainder 4} \quad \text{bit 1}
\end{align*}
\]

Write the hexadecimal number from bit 0 to bit 1 or right to left. The answer is \(77_{10} = 4D_{16}\).

(d) Convert the hexadecimal A6F number to decimal number.

\[
\begin{align*}
A \times 16^2 + 6 \times 16^1 + F \times 16^0 \\
= 10 \times 256 + 6 \times 16 + 15 \\
= 2560 + 96 + 15 \\
= 2671
\end{align*}
\]

The answer is \(A6F_{16} = 2671_{10}\).

(e) Write the hexadecimal equivalent of 1110111

\[
\begin{align*}
1110111 &= 111 \ 0111 \\
&= 7 \ 7
\end{align*}
\]

The answer is \(1110111_{16} = 77_{16}\).
10. (20 points) Assuming we have 4 projects and 2 exams for the CMSC104 class. Our grading methods are as follows:

- Projects: 60%
- Exams: 40%

Write a pseudocode to compute a student’s final grade for CMSC104 in terms of student’s name, grades for the projects and exams. Your pseudocode need to output student’s name and final grade. Your pseudocode should be clearly written and include 4 steps that we discussed in the class in order to get full credit.

**Solution:**
We assume all projects and exams have equal weight. To write pseudocode we do the following

- Define variables.
- Read the input.
- Compute the final grade.
- Output result.

So here is the pseudocode.

```plaintext
// Define variables
// Let
// name = name of a student
// projGrade = project grade
// examGrade = exam grade
// projectGrades = total project grades
// examGrades = total exam grades
// finalGrade = weighted average of projects and exams.

// Read input and compute total grade for each area.
Display "Enter student’s name: "
Read name
Display new_line

For (i = 1 to 4)
    Display "Enter Project Grade: "
    Read projGrade
    projectGrades = projectGrades + projGrade
End_For
Display new_line

For (i = 1 to 2)
    Display "Enter Exam Grade: "
    Read examGrade
```

3
examGrades = examGrades + examGrade
End_For
Display new_line

// Compute final grade
finalGrade = (0.6 * projGrades/4) + (0.4 * examGrades/2)

// Output result.
Display "Student’s name is 
Display name
Display new_line
Display "Student’s grade is 
Display finalGrade

11. (20 points) Give an algorithm to display an outline of right triangle whose height and width are both equal to an integer length \( L \). Error checking must be done for input \( L \), where \( L \geq 2 \). See figure below for an example of \( L = 5 \).

```
*****
* *
* *
**
*
```

**Solution:**
Here is how to display an outline of a right triangle whose height and width are both equal to an integer \( L \).

```
// Display first line
For i = 1 to L
  Display "*"
End_For
Display new_line

// Display 2nd to (last - 1) lines
For i = 1 to L - 2
  Display "*"
  For j = 1 to (L - 2 - i)
    Display " 
  End_For
  Display "*"
  Display new_line
End_For

// Display last line
12. (Extra Credit Problem: 10 points)
Given a real number $x$, ceiling of $x$, denoted $\lceil x \rceil$, is the smallest integer that is greater than or equal to $x$. For example, $\sqrt{6} = 2.4495$, and $\lceil 2.4495 \rceil = 3$.

Give an algorithm which inputs an integer $x$, where $x \geq 0$, computes and outputs $\lceil \sqrt{x} \rceil$.

Your algorithm does not need to be efficient, but it is not allowed to call the system’s build-in SQRT function.

Hint: Use the similar technique that we discussed for the $\lfloor \sqrt{x} \rfloor$ in the class.

Solution:
To write an algorithm or pseudocode, we do the following

- Define variables.
- Read the input.
- Compute ceiling of square root of $x$.
- Output results.

So here is an algorithm for computing and outputs $\lceil \sqrt{x} \rceil$, where $x \geq 0$.

```plaintext
// Define variables.
// Let x be an integer must be >= 0.
// Let y be ceiling of square root of x

// Read the input.
Display "Enter input integer x: "
Read x
While (x < 0)
    Display "Enter an integer x, where x must be >= 0: "
    Read x
End_While

// Compute ceiling of square root of x.
i = 0
While (i*i < x)
    i = i + 1
End_While
y = i

// Output results
Display "ceiling of square root of "
Display x
Display " is "
Display y
```