Designing Algorithms

Comp-110 Recitation
Sep 2, 2011

Review

• Algorithm
  – A step by step sequence of instructions for solving a problem
  – In a finite amount of time.

• Representing algorithms
  – Flow Chart
  – Pseudo Code
Problem: Eat potato chips

• Pseudo-Code

1. Take a packet full of chips
2. Open the packet
3. Repeat
   • Eat chips
   • Until packet is empty OR you are full.

Flow Chart Elements
**Example 1**

- **Problem:** Convert a binary number into decimal

- **Binary number**
  - Base 2
  - Uses only 0 and 1
  - Ex: 1010, 111, 10101 etc.

- **Decimal number**
  - Base 10
  - Uses digits 0 – 9
  - Ex: 10, 7, 21 etc.
Example 1

<table>
<thead>
<tr>
<th>Positions</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Number</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>$2^4$</td>
<td>$2^3$</td>
<td>$2^2$</td>
<td>$2^1$</td>
<td>$2^0$</td>
</tr>
<tr>
<td>Decimal Number</td>
<td>16</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Algorithm: Pseudo-Code

- Enter a binary number
- Initialize `decimal_number` to 0
- Pick the binary digit `bit` at `position = 0`
- Repeat until all bits are picked
  - Raise 2 to the power `position` and multiply by `bit`.
  - Add the result to `decimal_number`
  - Pick the `bit` at next `position`
- Print `decimal_number`
Algorithm: Flow Chart

Example 2

- Problem: Print factorial of a number

- Factorial calculation:
  - If n=0, result = 1.
  - If n>=1, result = n*(n-1)*(n-2)....*3*2*1.

Example:
factorial of 5 = 5 * 4 * 3 * 2 * 1 = 120
Algorithm: Pseudo-Code

1. Enter any positive integer \( n \)
2. Initialize \( \text{result} \) to 1.
3. Repeat until \( n \) is greater than 1.
   3.1 Multiply \( n \) to \( \text{result} \).
   3.2 Decrement \( n \) by 1.
4. Print \( \text{result} \).

Algorithm: Flow Chart

Start

Get any positive integer \( n \)

Initialize \( \text{result} \) to 1

Is \( n \geq 1 \)?

Yes

\( \text{result} = \text{result} \times n \)

No

Print \( \text{result} \)

Stop

Decrement \( n \) by 1
Exercise

• Problem: Find the largest among three integers.

• Write the algorithm and flow chart.

Algorithm: Pseudo-Code

• Enter three integers
• Read the integers \( a, b \) and \( c \).
• Initialize \( \text{largest} = a \)
• If \( b > \text{largest} \) then assign \( b \) to \( \text{largest} \).
• If \( c > \text{largest} \) then assign \( c \) to \( \text{largest} \).
• Print \( \text{largest} \).
Algorithm: Flow Chart

Start

Get three integers a, b and c

Initialize largest to a

Is b > largest

Yes

largest = b

No

Is c > largest

Yes

largest = c

No

Print largest

Stop