Data types and their representation

Jordi Cortadella
Department of Computer Science

Outline

• Data representation
• Boolean expressions
• Real numbers

The memory

How much memory do we have?

Our laptop has few Gbytes of memory:

1 Kilo (K) = 2^{10} = 1,024
1 Mega (M) = 2^{20} = 1,048,576
1 Giga (G) = 2^{30} = 1,073,741,824
1 Tera (T) = 2^{40} = 1,099,511,627,776
Write the binary representation

Design a procedure that, given a number \( n \), writes its binary representation (in reverse order).

```c
void base2(int n) {
    while (n > 1) {
        cout << n%2;
        n = n/2;
    }
    cout << n << endl;
}
```

Exercise

Design a procedure that, given a number \( n \) and a base \( b \), writes the representation of \( n \) in base \( b \) (in reverse order).

```c
void base(int n, int b);
```
Boolean expressions

Maximum of three numbers

// Returns max(a, b, c)
int max3(int a, int b, int c);

if (a > b and a > c) {
    return a;
} else {
    if (b > c) {
        return b;
    } else {
        return c;
    }
}
Maximum of three numbers

// Returns max(a, b, c)
int max3(int a, int b, int c) {
    if (a > b and a > c) return a;
    if (b > c) return b;
    return c;
}

Boolean operators

if (a > b and a > c) ...  
while (i >= 10 or c == 0) ...  
if (not (a < b and a < c)) ...

Truth tables

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>a and b</th>
</tr>
</thead>
<tbody>
<tr>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>a</th>
<th>not a</th>
</tr>
</thead>
<tbody>
<tr>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
</tr>
</tbody>
</table>

Boolean expressions

x > 2 and x < 7 \equiv x >= 3 and x <= 6

x <= 2 or x > 7

x <= 2 not (x <= 2) \equiv x > 2

x == 5
**Boolean expressions**

\[ x > 2 \quad \text{and} \quad x < 7 \equiv x \geq 3 \quad \text{and} \quad x \leq 6 \]

\[ x \leq 2 \quad \text{or} \quad x > 7 \]

\[ x \leq 2 \quad \text{not} \ (x \leq 2) \equiv x > 2 \]

\[ \text{not} \ (x == 5) \equiv x \neq 5 \]

\[ x == 0 \quad \text{or} \quad x == 6 \]

\[ x \% 2 == 0 \]

**Exercise**

\[ x == 0 \quad \text{or} \quad x > 5 \]

\[ x \% 2 == 0 \]
**Exercise**

\[ (x > -1 \text{ and } x < 3) \text{ or } (x \geq 6 \text{ and } x < 11) \]

**Complement of and/or**

\[ (x > 2 \text{ and } x < 7) \]
\[ \text{not} (x > 2 \text{ and } x < 7) \]
\[ x \leq 2 \text{ or } x \geq 7 \]

**De Morgan’s law**

\[ \text{not} (e_1 \text{ and } e_2) \equiv \text{not } e_1 \text{ or } \text{not } e_2 \]
\[ \text{not} (e_1 \text{ or } e_2) \equiv \text{not } e_1 \text{ and } \text{not } e_2 \]

**Operator precedence**

\[ a + b \times c \]
\[ (a + b) \times c \]
\[ a \text{ or } b \text{ and } c \]
\[ (a \text{ or } b) \text{ and } c \]

**Exercise**

Simplify: \[ \text{not} (x \geq y \text{ or } y \% 2 == 0) \]

\[ x < y \text{ and } y \% 2 != 0 \]
Real numbers

Circles intersect if and only if the distance between centers is smaller than or equal to the sum of radii.

\[ \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \leq r_1 + r_2 \]

\[ (x_1 - x_2)^2 + (y_1 - y_2)^2 \leq (r_1 + r_2)^2 \]

Intersection of circles

- Write a program that reads the center and the radius of two circles and prints “yes” or “no” depending on whether they intersect or not.

- Example:

  - Center 1: \((x_1, y_1, r_1) = (2, 5.3, 1.34)\)
  - Center 2: \((x_2, y_2) = (0, 0, 2)\)
  - Intersection: no

  - Center 1: \((x_1, y_1) = (1.5, 2.5, 10)\)
  - Center 2: \((x_2, y_2) = (0.5, 3.6, 4.3)\)
  - Intersection: yes
// Reads the center and radius of two circles
// and prints whether they intersect or not.
int main() {
    double x1, y1, r1, x2, y2, r2; // Real numbers
    cin >> x1 >> y1 >> r1 >> x2 >> y2 >> r2;

    double dx = x1 - x2; // (x1 - x2)^2
    double dy = y1 - y2; // (y1 - y2)^2
    double r = r1 + r2; // (r1 + r2)^2

    bool intersect = dx + dy <= r; // true or false

    if (intersect) cout << "yes" << endl;
    else cout << "no" << endl;
}

Real numbers

• Two types:
  – float (single precision, 32 bits)
  – double (double precision, 64 bits)

• Arithmetic operations: + - * /  (no remainder)

• Real constants:
  2  -5.003  3.1416  1.4e9  0.6E-15
     \biggarrow
  1.4 \cdot 10^9  0.6 \cdot 10^{-15}

Type conversion

• Arithmetic operations between integer and real values usually imply an implicit type conversion.
• Be careful:

    int i=3, j=2;
    double x;
    x = i/j; // x = 1.0
    x = i/\textit{double}(j); // x = 1.5
    x = \textit{double}(i)/j; // x = 1.5
    x = \textit{double}(i/j); // x = 1.0
    x = i/2; // x = 1.0
    x = i/2.0; // x = 1.5

    i = x; // i = 1
    j = 3.14159265; // j = 3
// Returns $x^2$
double sq(double x) {
    return x * x;
}

// Reads the center and radius of two circles
// and prints whether they intersect or not.
int main() {
    double x1, y1, r1, x2, y2, r2;
    cin >> x1 >> y1 >> r1 >> x2 >> y2 >> r2;

    if (sq(x1 - x2) + sq(y1 - y2) <= sq(r1 + r2)) cout << "yes";
    else cout << "no";

    cout << endl;
}