

---

Subject: String near match algorithm

Posted by [Didier](#) on Fri, 31 Jul 2009 19:00:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

For data importing I have written a small algorithm intended to compare 2 strings and indicate if the two strings are close to each other or not.

I used it to avoid importing into a DB data with misspelled names

```
int correlation(const String& a, const String& b)
{
    int res=0;
    const char* A = a.Begin();
    const char* B = b.Begin();

    const int Al = a.GetLength();
    const int Bl = b.GetLength();
    const int deltaL = Al - Bl;

    int As, Bs, intersectLength;
    int subRes;

    int matchPatternMinLength = max(2, min(b.GetLength(), a.GetLength())/3);

    for (int offset = (matchPatternMinLength-Bl); offset <= (Al-matchPatternMinLength); offset++)
    {
        // range calculation
        if (offset < 0)
        {
            As = 0;
            Bs = -offset;
            if (offset < deltaL) intersectLength = offset + Bl;
            else                intersectLength = Al;
        }
        else
        {
            As = offset;
            Bs = 0;
            intersectLength = Al - offset;
            if (offset <= deltaL) intersectLength = Bl;
            else                intersectLength = Al-offset;
        }

        subRes = 0;
        for (int c = 0; c<intersectLength; c++)
        {
```

```

    if( A[As+c] == B[Bs+c]) ++subRes;
  }
  if (subRes >= matchPatternMinLength) res += subRes;
}

```

```

// taking string length in account
if ( deltaL > 0 )
{
  res -= deltaL;
}
else
{
  res += deltaL;
}
return res;
}

```

```

inline bool CompareDistance(const String& a, const String& b)
{
  if (correlation(a, b) >= max(2, min(a.GetLength(), b.GetLength())*3/5)) return true;
  return false;
}

```

The algorithm is based on a basic signal processing technique like in sonars or radars (correlation) adapted to strings.  
 It is not very optimized but it works fine. The function `CompareDistance()` is only here to introduce a threshold value: 3/5 in the code.

How to use it:

If `CompareDistance(stringA, stringB)` returns true, then the two strings are considered "near match" ==> maybe misspelled names.

Maybe this can be useful to somebody ( if it doesn't exist anywhere else ) .