**Fibonacci Search**

*(invented in 1960 by D.E. Ferguson)*

**// Fibonacci search, search A[1],A[2],...,A[N]**

**// Here N+1 must be a Fibonacci number, F[m+1]**

**int fibSearch(int y, A[1..N])**

**{ // y = A[x], 0 < x < F[m+1]**

**int mid = F[m]; // F[m] is partway from 0 to F[m+1]**

**int p = F[m-1] = F[m] - F[m-1]**

**int q = F[m-2] = 2\*F[m-1} - F[m]**

**for (;;)**

**{**

**if (y == A[mid])**

**return mid;**

**else if (y < A[mid])**

**{**

**if (q == 0)**

**return -(mid - 1);**

**mid = mid - q;**

**int temp = p;**

**p = q;**

**q = temp - q;**

**}**

**else if (y > A[mid])**

**{**

**if (p == 1)**

**return -mid;**

**mid = mid + q;**

**p = p - q;**

**q = q - p;**

**}**

**}**

**}**