```
array-count-update  THM

⊢  ∀T:Type, P:(T→𝔹), a:array(T), i:ℕ|a|, v:T.
|  array-count(v.P(v);a[i:=v])
|  =
|  array-count(v.P(v);a)+if
|                         P(v)→if P(a[i])→ 0 else 1 fi
|                        ;P(a[i])→-1
|                        else 0 fi
| by Auto THEN (Unfold ‘array-count‘ 0) THEN (Reduce 0)
|    THEN
|    (Inst
|     Thm* ∀n:ℕ, f:(ℕn→ℤ), m:ℕn.
|          sum(f(x) | x < n)
|          =
|          f(m)+sum(if x=₂m→ 0 else f(x) fi | x < n)
|     {isolate_summand}
|     [|a|;λx.if P(a[i:=v][x])→ 1 else 0 fi;i])
|    THEN
|    (All ReduceSOAps)
|    THEN
|    (HypSubst -1 0)
|    THEN
|    (Thin -1)
|    THEN
|    (Inst
|     Thm* ∀n:ℕ, f:(ℕn→ℤ), m:ℕn.
|          sum(f(x) | x < n)
|          =
|          f(m)+sum(if x=₂m→ 0 else f(x) fi | x < n)
|     {isolate_summand}
|     [|a|;λx.if P(a[x])→ 1 else 0 fi;i])
|    THEN
|    (All ReduceSOAps)
|    THEN
|    (HypSubst -1 0)
|    THEN
|    (Thin -1)
|    THEN
|    (Subst (a[i:=v][i] ~ v) 0)
|
|\
| .....equality .....
| 1. T : Type
| 2. T→𝔹
| 3. a : array(T)
| 4. i : ℕ|a|
| 5. v : T
| ⊢  a[i:=v][i] ~ v
|   by (RWO
|       Thm* ∀a:array(T), j,i:ℕ|a|, v:T.
|            a[i:=v][j] ~ if j=₂i→ v else a[j] fi
|       {array_DASH_update_DASH_select}
|       0)
|      THEN
```

```
|      SplitOnConclITE
|
.\
. 1. T : Type
. 2. P : T→𝔹
. 3. a : array(T)
. 4. i : ℕ|a|
. 5. v : T
. ⊢  if P(v)→ 1 else 0 fi
.    +sum(if x=₂i→0 ;P(a[i:=v][x])→1 else 0 fi | x < |a|)
.    =
.    if P(a[i])→ 1 else 0 fi
.    +sum(if x=₂i→0 ;P(a[x])→1 else 0 fi | x < |a|)
.    +if P(v)→if P(a[i])→ 0 else 1 fi ;P(a[i])→-1 else 0 fi
.   by (AssertBY
.       (sum(if x=₂i→0 ;P(a[i:=v][x])→1 else 0 fi | x < |a|)
.        =
.        sum(if x=₂i→0 ;P(a[x])→1 else 0 fi | x < |a|))
.       ((BackThru
.         Thm* ∀n:ℕ, f,g:(ℕn→ℤ).
.              (∀i:ℕn. f(i) = g(i))
.              ⇒
.              sum(f(x) | x < n) = sum(g(x) | x < n)
.         {sum_functionality})
.        THEN
.        (RWO
.         Thm* ∀a:array(T), j,i:ℕ|a|, v:T.
.              a[i:=v][j] ~ if j=₂i→ v else a[j] fi
.         {array_DASH_update_DASH_select}
.         0)
.        THEN
.        SplitOnConclITE
.        THEN
.        (Reduce 0)))
.      THEN
.      (RepeatFor 2 SplitOnConclITE)
END
```