

Semantyka i weryfikacja programów 2022/23.
Egzamin 21/02/2023, zadanie 3 (weryfikacja)

Pracujemy w języku $\text{TINY}_{\mathcal{A}}$ nad typem danych rozszerzonym o rodzaj Array i operacje:

$\text{newarr} : \text{Array}$
 $\text{put} : \text{Array} \times \text{Int} \times \text{Int} \rightarrow \text{Array}$
 $\text{get} : \text{Array} \times \text{Int} \rightarrow \text{Int}$
 $\text{swap} : \text{Array} \times \text{Int} \times \text{Int} \rightarrow \text{Array}$

Nośnik rodzaju Array to zbiór funkcji (całkowitych) z liczb całkowitych w liczby całkowite,

$$|\mathcal{A}|_{\text{Array}} = \mathbf{Int} \rightarrow \mathbf{Int},$$

a operacje interpretowane są jako funkcje

$\text{newarr}_{\mathcal{A}} : |\mathcal{A}|_{\text{Array}}$
 $\text{put}_{\mathcal{A}} : |\mathcal{A}|_{\text{Array}} \times |\mathcal{A}|_{\text{Int}} \times |\mathcal{A}|_{\text{Int}} \rightarrow |\mathcal{A}|_{\text{Array}}$
 $\text{get}_{\mathcal{A}} : |\mathcal{A}|_{\text{Array}} \times |\mathcal{A}|_{\text{Int}} \rightarrow |\mathcal{A}|_{\text{Int}}$
 $\text{swap}_{\mathcal{A}} : |\mathcal{A}|_{\text{Array}} \times |\mathcal{A}|_{\text{Int}} \times |\mathcal{A}|_{\text{Int}} \rightarrow |\mathcal{A}|_{\text{Array}}$

zdefiniowane następująco:

$\text{newarr}_{\mathcal{A}}(i) = 0$
 $\text{put}_{\mathcal{A}}(A, i, n) = A[i \mapsto n]$
 $\text{get}_{\mathcal{A}}(A, i) = A(i)$
 $\text{swap}_{\mathcal{A}}(A, i, j) = A[i \mapsto A(j), j \mapsto A(i)]$

dla wszystkich $i, j, n \in \mathbf{Int}$ i $A : \mathbf{Int} \rightarrow \mathbf{Int}$. Wyrażenie $\text{get}(A, e)$ będziemy jak zwykle zapisywać jako $A[e]$. Ponadto, w asercjach wykorzystujemy „predykat” $A : \text{Array}$ stwierdzający, że zmienna A jest rodzaju Array (pozostałe zmienne są rodzaju Int). Wyrażenia logiczne języka rozszerzamy też o oczywiste nierówności $e < e'$ (wyrażalne jako $e \leq e' \wedge \neg(e = e')$) oraz $e > e'$ (wyrażalne jako $e' < e$). Dla czytelności, w formułach ciągi nierówności będziemy zapisywać w skróconej postaci, np. $j \leq i \wedge i < k$ zapisując jako $j \leq i < k$.

Dla czytelności, rozszerzamy syntaksę języka o nawiasy $\text{begin} \dots \text{end}$.

Dla wygody specyfikacji i dowodzenia poprawności wprowadzamy następujące schematy formuł:

$$\begin{aligned} S^+(a, j, k) &\equiv a : \text{Array} \wedge \forall i. (j \leq i < k \implies a[i] \leq a[i+1]) \\ S^-(a, j, k) &\equiv a : \text{Array} \wedge \forall i. (j \leq i < k \implies a[i] \geq a[i+1]) \\ G(a, i, j) &\equiv \exists g. i \leq g \leq j \wedge S^+(a, i, g) \wedge S^-(a, g, j) \end{aligned}$$

Należy udowodnić całkowitą poprawność następującego programu względem podanych warunków:

```

[a:Array ∧ G(a,1,n) ∧ n>0 ]
j:=n;
while (j>0) do
begin
  if (a[1]≥a[j]) then j:=j-1
  else
  begin
    a[0]:=a[1];
    i:=1;
    while i<j do
    begin
      a[i]:=a[i+1];
      i:=i+1;
    end;
    a[j]:=a[0];
    j:=j-1;
  end
end
[S-(a,1,n)]

```

Na następnej stronie podana jest wersja tego programu z miejscem na wpisanie odpowiednich asercji, którą można wykorzystać dla przedstawienia rozwiązania .

Wymagane jest:

- podanie niezmienników γ_1 i γ_2 obu pętli programu oraz asercji $\alpha_1, \alpha_2, \alpha_3, \alpha_4$, które „koduja” dowód poprawności (podanie pozostałych asercji jest opcjonalne, ale jeśli zostaną podane, to ewentualne błędy mogą wpłynąć na ocenę rozwiązania), oraz
- podanie anotacji [**decr ...in ...wrt ...**] tak, aby (w kontekście podanych niezmienników) wynikała z nich własność stopu pętli.

```

[a:Array ∧ G(a,1,n) ∧ n>0 ]
j:=n;
[
while (j>0) do [γ1:
]
    [decr ...in ...wrt ...]
    begin
[
    if (a[1]≥a[j]) then
[ α1:
    j:=j-1;
    else
    begin
[α2:
    a[0]:=a[1];
[
    i:=1;
[
    while i<j do [γ2:
]
        [decr ...in ...wrt ...]
        begin
[
            a[i]:=a[i+1];
[
            i:=i+1;
[
            end;
[α3:
[
            a[j]:=a[0];
[α4:
            j:=j-1;
[
        end
[
    end
[
end
[
[S-(a,1,n)]

```

Mozliwe rozwiązanie:

```

[a:Array ∧ G(a,1,n) ∧ n>0 ]
[S-(a,n,n) ∧ G(a,1,n) ∧ n<n =>(a[j+1]≤a[1])
  ∧ n>0 ∧ 0≤n≤n]

j:=n;
[S-(a,j,n) ∧ G(a,1,j) ∧ j<n =>(a[j+1]≤a[1]) ∧ n>0 ∧ 0≤j≤n ]
  while (j>0) do [γ1: S-(a,j,n) ∧ G(a,1,j) ∧ j<n =>(a[j+1]≤a[1])
    ∧ n>0 ∧ 0≤j≤n]
    [decr j in Nat wrt <]

  begin
    if (a[1]≥a[j]) then
[γ1 ∧ a[1]≥a[j] ∧ j>0 ]
[α1: S-(a,j-1,n) ∧ G(a,1,j-1) ∧ j-1<n =>(a[j]≤a[1])
  ∧ n>0 ∧ 0≤j-1≤n]

      j:=j-1;
    else
      begin
[γ1 ∧ a[1]<a[j] ∧ j>0 ]
[α2: S-(put(a,0,a[1]),j,n) ∧ G(put(a,0,a[1]),1,j)
  ∧ put(a,0,a[1])[0]<put(a,0,a[1])[j]
  ∧ j<n =>(put(a,0,a[1])[j+1]≤put(a,0,a[1])[0]) ∧ put(a,0,a[1])[0]≤put(a,0,a[1])[1]
  ∧ (put(a,0,a[1])[0]=a[1] ∧ 1≤1≤j ∧ n>0 ∧ 0<j≤n ]
      a[0]:=a[1];
[ S-(a,j,n) ∧ G(a,1,j) ∧ a[0]<a[j]
  ∧ j<n =>(a[j+1]≤a[0]) ∧ a[0]≤a[1] ∧ a[0]=a[1] ∧ 1≤1≤j
  ∧ n>0 ∧ 0<j≤n ]
      i:=1;
[ S-(a,j,n) ∧ G(a,1,j) ∧ a[0]<a[j]
  ∧ j<n =>(a[j+1]≤a[0]) ∧ a[0]≤a[1] ∧ a[i-1]=a[i] ∧ 1≤i≤j
  ∧ n>0 ∧ 0<j≤n ]
      while i<j do [γ2: S-(a,j,n) ∧ G(a,1,j) ∧ a[0]<a[j]
        ∧ j<n =>(a[j+1]≤a[0]) ∧ a[0]≤a[1] ∧ a[i-1]=a[i] ∧ 1≤i≤j
        ∧ n>0 ]
        [decr j-i+1 in Nat wrt <]

      begin
[ S-(put(a,i,a[i+1]),j,n) ∧ G(put(a,i,a[i+1]),1,j)
  ∧ a[0]<put(a,i,a[i+1])[j] ∧ j<n-1 =>(put(a,i,a[i+1])[j+1]≤a[0])
  ∧ a[0]≤put(a,i,a[i+1])[0] ∧ put(a,i,a[i+1])[i]=put(a,i,a[i+1])[i+1]
  ∧ 1≤i+1≤j ∧ n>0 ∧ j<n ]
      a[i]:=a[i+1];
[ S-(a,j,n) ∧ G(a,1,j) ∧ a[0]<a[j]
  ∧ j<n =>(a[j+1]≤a[0]) ∧ a[0]≤a[1] ∧ a[i]=a[i+1] ∧ 1≤i+1≤j ∧ n>0 ∧ j≤n ]
      i:=i+1;
[ S-(a,j,n) ∧ G(a,1,j) ∧ a[0]<a[j]
  ∧ j<n =>(a[j+1]≤a[0]) ∧ a[0]≤a[1] ∧ a[i-1]=a[i] ∧ 1≤i≤j ∧ n>0 ∧ 0<j≤n ]

```

```

end;
[α3:  $S^-(a, j, n) \wedge G(a, 1, j) \wedge a[0] < a[j] \wedge j < n \Rightarrow (a[j+1] \leq a[0])$ 
 $\wedge a[0] \leq a[1] \wedge a[i-1] = a[i] \wedge 1 \leq i \leq j \wedge n > 0 \wedge 0 < j \leq n \wedge i = j]$ 

 $[S^-(\text{put}(a, j, a[0]), j-1, n) \wedge G(\text{put}(a, j, a[0]), 1, j-1) \wedge j-1 < n \Rightarrow (\text{put}(a, j, a[0])[j] \leq a[0])$ 
 $\wedge n > 0 \wedge 0 \leq j-1 < n]$ 
 $a[j] := a[0];$ 
[α4:  $S^-(a, j-1, n) \wedge G(a, 1, j-1) \wedge j-1 < n \Rightarrow (a[j] \leq a[1])$ 
 $\wedge n > 0 \wedge 0 \leq j-1 \leq n]$ 
 $j := j-1;$ 
 $[S^-(a, j, n) \wedge G(a, 1, j) \wedge j < n \Rightarrow (a[j+1] \leq a[1])$ 
 $\wedge n > 0 \wedge 0 \leq j \leq n]$ 
end
 $[S^-(a, j, n) \wedge G(a, 1, j) \wedge j < n \Rightarrow (a[j+1] \leq a[1])$ 
 $\wedge n > 0 \wedge 0 \leq j \leq n]$ 
end
 $[S^-(a, 1, n)]$ 

```