

Derivation of Algorithms Lab4 Solutions

Exercise from Guarded Command section of notes

Write guarded command programs for the following:

Read the lengths of three lines and determine if they form a triangle and, if so, print one of scalene, isosceles or equilateral.

```
|[ con A, B, C: int; {A > 0 ∧ B > 0 ∧ C > 0}
```

```
  var
```

```
  triangle : boolean;
```

```
  triangle := false;
```

```
  if A ≥ C ∧ A ≥ B →
```

```
    triangle = (B + C) ≥ A;
```

```
  [] B ≥ A ∧ B ≥ C →
```

```
    triangle = (A + C) ≥ B;
```

```
  [] C ≥ A ∧ C ≥ B →
```

```
    triangle = (A + B) ≥ C;
```

```
  fi;
```

```
  if triangle = true →
```

```
    if A=B ∧ B=C →
```

```
      // EQUILATERAL
```

```
    [] A≠B ∧ B≠C →
```

```
      // SCALENE
```

```
    [] A=B ∨ C=B ∨ C=A →
```

```
      // ISOSCELES
```

```
    fi;
```

```
  [] triangle ≠ true →
```

```
    // NOT A TRIANGLE
```

```
  fi;
```

```
]|
```

Initialise an array of size N to random values in the range 0..100 and compute the sum of the elements in the array.

```
[[ con N: int; {N > 0}
```

```
  var
```

```
  f: array[0..N) of int;
```

```
  n: int;
```

```
  sum: int;
```

```
  n:=0;
```

```
  do n < N →
```

```
    f.n = rand(100) + 1;
```

```
    n:=n+1;
```

```
  od;
```

```
  n, sum := 0,0;
```

```
  do n < N →
```

```
    sum := sum + f.n;
```

```
    n:=n+1;
```

```
  od;
```

```
]]
```

Write program specifications for the following

B is equivalent to X is a positive whole number

$[[\text{con } B: \text{boolean};$

var $x: \text{int};$

S

$\{B \equiv x > 1;\}$

$]]$

j is the index of the smallest value in f

$[[\text{con } N: \text{int}; \{N > 0\}$

f: array[0..N) of int;

var $j: \text{int};$

S

$\{0 \leq j < N \wedge \forall x: 0 \leq x < N: f.j \leq f.x\}$

$]]$