

1)

IGCSE pseudocode

```
DECLARE V1 : ARRAY[0:2] OF INTEGER
DECLARE V2 : ARRAY[0:2] OF INTEGER
DECLARE XV : ARRAY[0:2] OF INTEGER

OUTPUT "ENTER X COORDINATE FOR VECTOR 1"
INPUT V1X
OUTPUT "ENTER Y COORDINATE FOR VECTOR 1"
INPUT V1Y
OUTPUT "ENTER Z COORDINATE FOR VECTOR 1"
INPUT V1Z
OUTPUT "ENTER X COORDINATE FOR VECTOR 2"
INPUT V2X
OUTPUT "ENTER Y COORDINATE FOR VECTOR 2"
INPUT V2Y
OUTPUT "ENTER Z COORDINATE FOR VECTOR 2"
INPUT V2Z

V1[0] <- V1X
V1[1] <- V1Y
V1[2] <- V1Z

V2[0] <- V2X
V2[1] <- V2Y
V2[2] <- V2Z

XV[0] <- V1[1] * V2[2] - V1[2] * V2[1]
XV[1] <- V1[2] * V2[0] - V1[0] * V2[2]
XV[2] <- V1[0] * V2[1] - V1[1] * V2[0]

OUTPUT "THE CROSS PRODUCT OF VECTORS 1 AND 2 IS ", XV
```

Syntax Check

Run

igpc v0.0.3

CONSOLE

```
ENTER X COORDINATE FOR VECTOR 1
ENTER Y COORDINATE FOR VECTOR 1
ENTER Z COORDINATE FOR VECTOR 1
ENTER X COORDINATE FOR VECTOR 2
ENTER Y COORDINATE FOR VECTOR 2
ENTER Z COORDINATE FOR VECTOR 2
THE CROSS PRODUCT OF VECTORS 1 AND 2 IS [-3,6,-3]
```

Clear

2)

IGCSE pseudocode

```
DECLARE LIST : ARRAY [0:3] OF CHAR
DECLARE INDEX : INTEGER

LIST[0] <- 'h'
LIST[1] <- 'e'
LIST[2] <- 'l'
LIST[3] <- 'l'
LIST[4] <- 'o'
LIST[5] <- 'n'
LIST[6] <- 'i'
LIST[7] <- 'c'
LIST[8] <- 'k'

OUTPUT LIST

FOR INDEX <- 0 TO 8
  IF MOD(INDEX, 2) = 1
    THEN
      LIST[INDEX] <- UCASE(LIST[INDEX])
    ENDIF
  NEXT INDEX

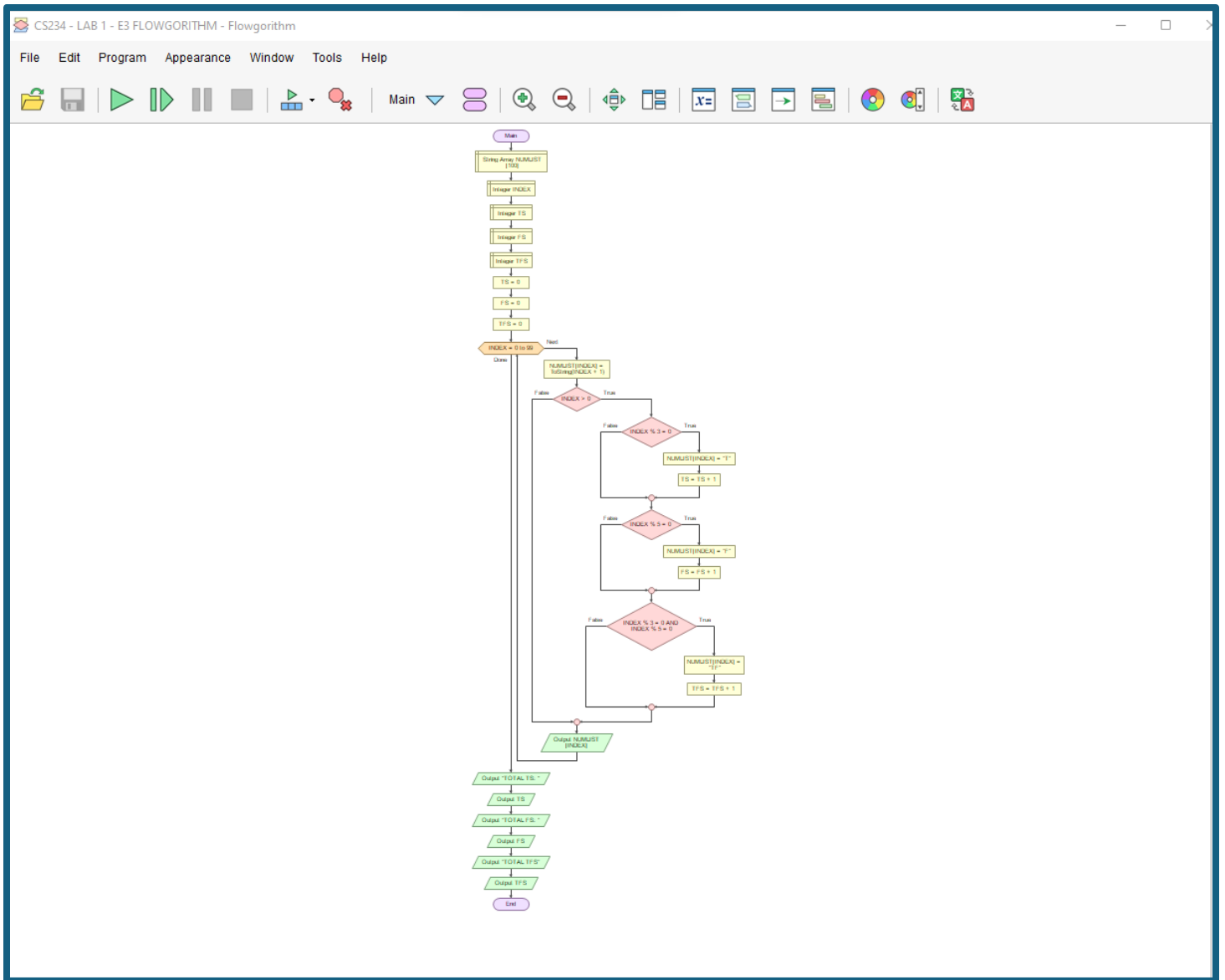
OUTPUT LIST
```

CONSOLE

```
["h","e","l","l","o","n","i","c","k"]
["h","E","l","L","o","N","i","C","k"]
```

Clear

3)





87

T

89

90

TF

92

93

T

95

F

T

98

99

T

TOTAL TS:

33

TOTAL FS:

19

TOTAL TFS

6

IGCSE pseudocode

```
DECLARE NUMLIST : ARRAY [0:99] OF CHAR
DECLARE INDEX : INTEGER
TS <- 0
FS <- 0
TFS <- 0

FOR INDEX <- 0 TO 99
    NUMLIST[INDEX] <- INDEX + 1
    IF INDEX > 0
        THEN
            IF MOD(INDEX + 1, 3) = 0
                THEN
                    NUMLIST[INDEX] <- "T"
                    TS <- TS + 1
                ENDIF
            IF MOD(INDEX + 1, 5) = 0
                THEN
                    NUMLIST[INDEX] <- "F"
                    FS <- FS + 1
                ENDIF
            IF MOD(INDEX, 3) = 0 AND MOD(INDEX, 5) = 0
                THEN
                    NUMLIST[INDEX] <- "TF"
                    TFS <- TFS + 1
                ENDIF
            ENDIF
        OUTPUT NUMLIST[INDEX]
    NEXT INDEX

OUTPUT "TOTAL Ts: ", TS
OUTPUT "TOTAL Fs: ", FS
OUTPUT "TOTAL TFs: ", TFS
```

CONSOLE

```
1
2
T
4
F
T
7
8
T
F
11
T
13
14
F
TF
17
T
19
F
T
22
23
T
F
26
T
28
29
F
TF
32
T
34
F
T
37
38
T
F
41
T
43
```

```
79
F
T
82
83
T
F
86
T
88
89
F
TF
92
T
94
F
T
97
98
T
F
TOTAL Ts: 33
TOTAL Fs: 20
TOTAL TFs: 6
```

Clear

4)

IGCSE pseudocode

```
DECLARE HEIGHT : INTEGER
DECLARE WEIGHT : INTEGER
DECLARE BMI : REAL

OUTPUT "PLEASE ENTER HEIGHT IN INCHES AND WEIGHT IN POUNDS"
INPUT HEIGHT
INPUT WEIGHT

BMI <- 703 * (WEIGHT / (HEIGHT * HEIGHT))

IF BMI <= 18.5
  THEN
    OUTPUT "BMI: ", BMI, " -UNDERWEIGHT"
  ENDF

IF BMI >= 18.5 AND BMI <= 24.9
  THEN
    OUTPUT "BMI: ", BMI, " -HEALTHY WEIGHT"
  ENDF

IF BMI >= 25 AND BMI <= 29.9
  THEN
    OUTPUT "BMI: ", BMI, " -OVERWEIGHT"
  ENDF

IF BMI >= 30
  THEN
    OUTPUT "BMI: ", BMI, " -OBESITY"
  ENDF
```

Syntax Check

Run

igpc v0.0.3

CONSOLE

```
PLEASE ENTER HEIGHT IN INCHES AND WEIGHT IN POUNDS
BMI: 24.853535353535353 -HEALTHY WEIGHT
```

Clear

