

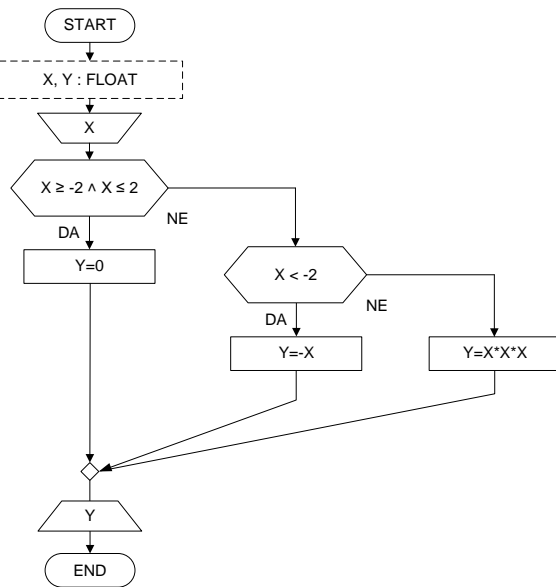
Računske vježbe 1

1. Kreirati algoritam i napisati pseudokod za izračunavanje vrijednosti **Y** u zavisnosti od broja **X**, koji predstavlja ulazni podatak.

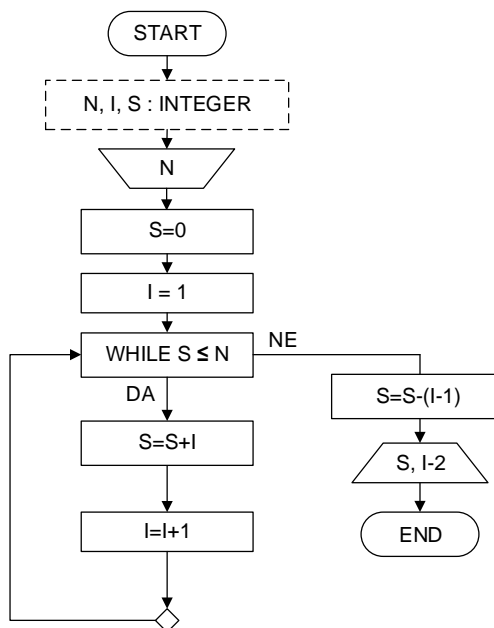
$$Y = \begin{cases} 0 & |X| \leq 2 \\ -X & X < -2 \\ X^3 & X > 2 \end{cases}$$

```

X, Y: FLOAT
INPUT X
IF X ≥ -2 ^ X ≤ 2
Y = 0
ELSEIF X < -2
Y = -X
ELSE
Y = X * X * X
ENDIF
OUTPUT Y
END
    
```



2. Na osnovu date algoritamske šeme napisati pseudokod i odrediti šta će biti izlaz algoritma za unijeto **N=7**.



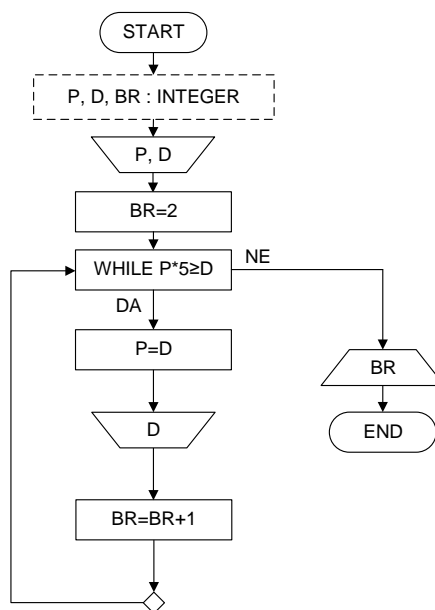
3. Na osnovu datog pseudokoda kreirati algoritamsku šemu i odrediti izlaz algoritma za $N=5$.

```

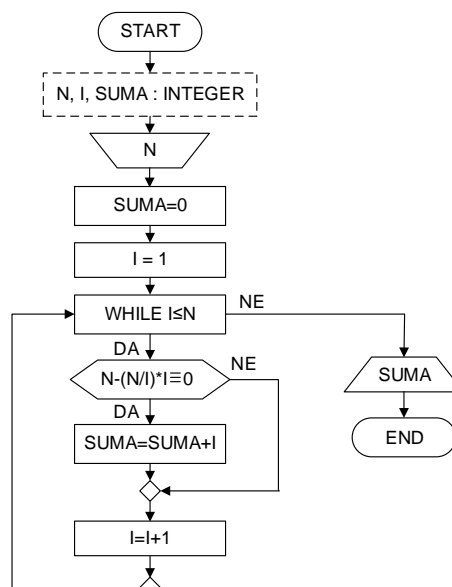
N, FAKT: INTEGER
INPUT N
FAKT=1
WHILE N≥1
FAKT=FAKT*N
N=N-1
ENDWHILE
OUTPUT FAKT
END
    
```

4. Kreirati algoritam kojim se vrši unos cijelih brojeva sve dok se ne unese broj koji je bar 5 puta veći od prethodno unesenog broja. Odštampati koliko je brojeva uneseno.

Primjer: Nakon unosa brojeva 2, 8, 3, 7, 4, 22, program štampa broj 6 (22 je bar 5 puta veći od 4)

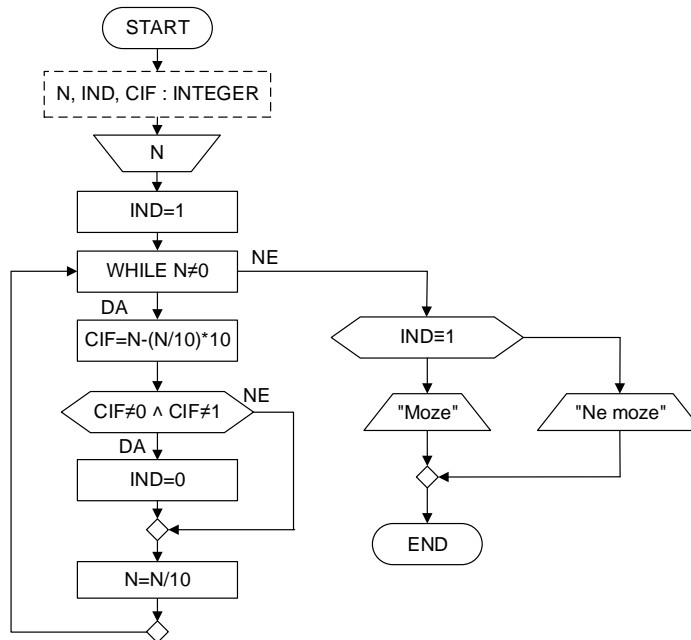


5. Kreirati algoritam i napisati pseudokod koji za unijeti prirodan broj N računa sumu svih prirodnih brojeva sa kojima je N djeljiv.



6. Kreirati algoritam kojim se učitava prirodan broj N i koji provjerava da li taj broj može predstavljati binarni zapis nekog broja, tj. da li su sve cifre 1 ili 0. Štampati odgovarajuću poruku na izlazu.

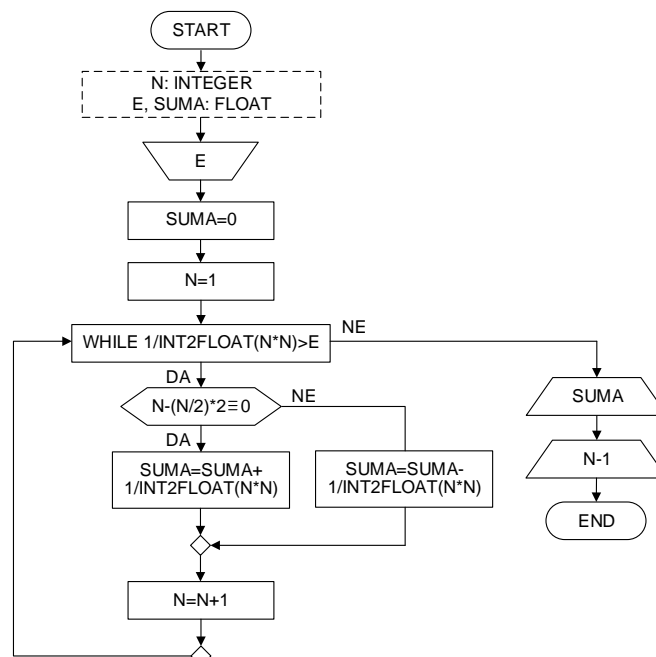
Primjer: Broj N=101101 može predstavljati binarni zapis nekog broja, dok N=127 ne može.



7. Kreirati algoritam koji približno računa sumu reda:

$$S = \sum_{n=1}^{\infty} (-1)^n \frac{1}{n^2}$$

Sumiranje vršiti sve dok je apsolutna vrijednost opšteg člana sume veća od zadate tačnosti E, koja se unosi na početku algoritma. Na izlazu ispisati dobijenu vrijednosti i broj sumiranih članova.



8. Kreirati algoritam kojim se učitava cijeli broj **N** i koji štampa n-ti član Fibonačijevog niza. Fibonačijev niz predstavlja niz brojeva u kome zbir prethodna dva broja u nizu daje vrijednost narednog člana niza. Prva dva člana su mu 0 i 1.

Primjer: Za **N=9** algoritam će odštampati 34 jer je niz 0, 1, 1, 2, 3, 5, 8, 13, 21, 34... gdje je $F_0=0$, $F_1=1$ a $F_9=34$.

```
N, F, P, D, BR, I: INTEGER  
INPUT N  
P=0, D=0, F=1, BR=3  
FOR I=BR:N+1  
F=P+D  
P=D  
D=F  
ENDFOR  
OUTPUT F  
END
```