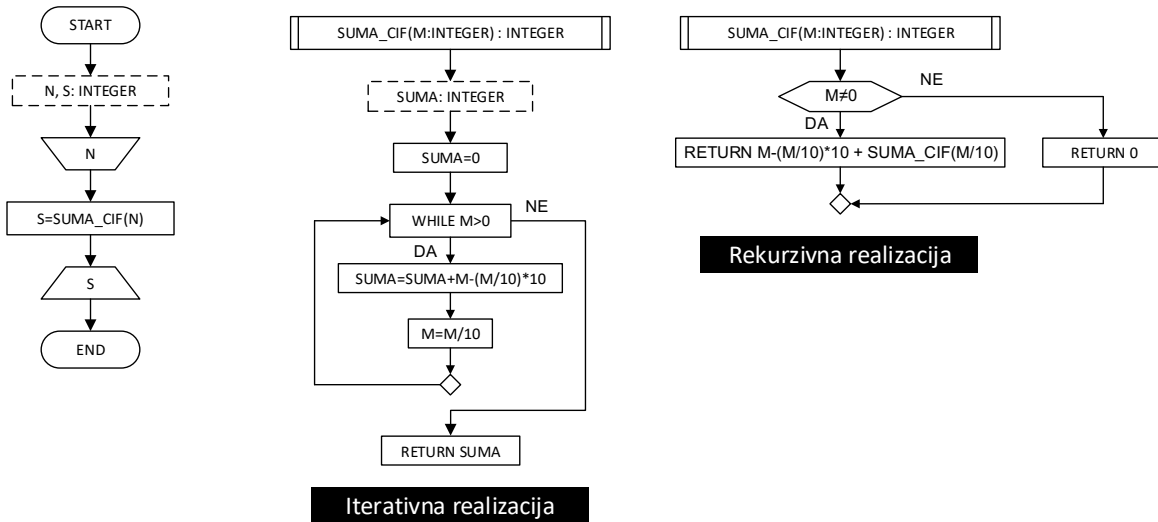
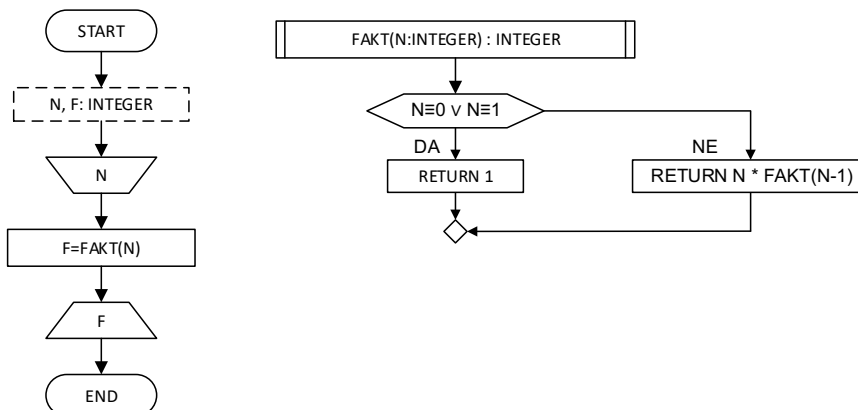


Računske vježbe 8

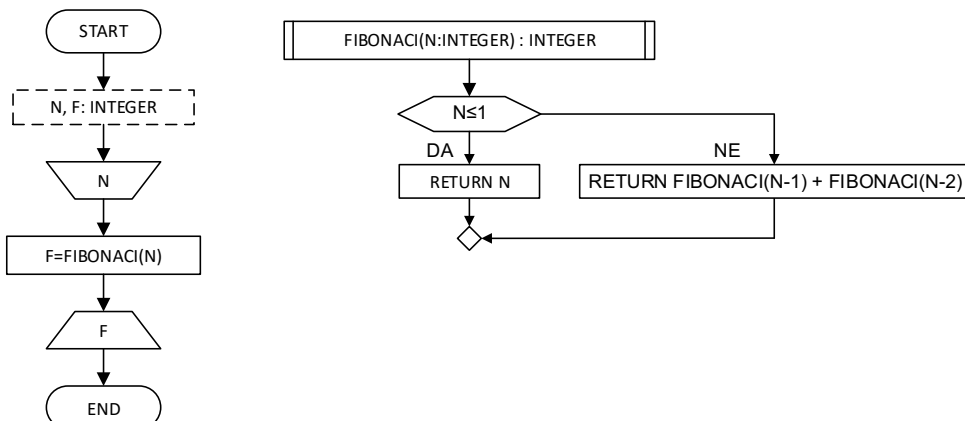
1. Kreirati algoritam kojim se učitava prirodan broj **N** i koji računa sumu njegovih cifara pomoću rekurzivne funkcije **SUMA_CIF**. U glavnom programu štampati dobijenu sumu.



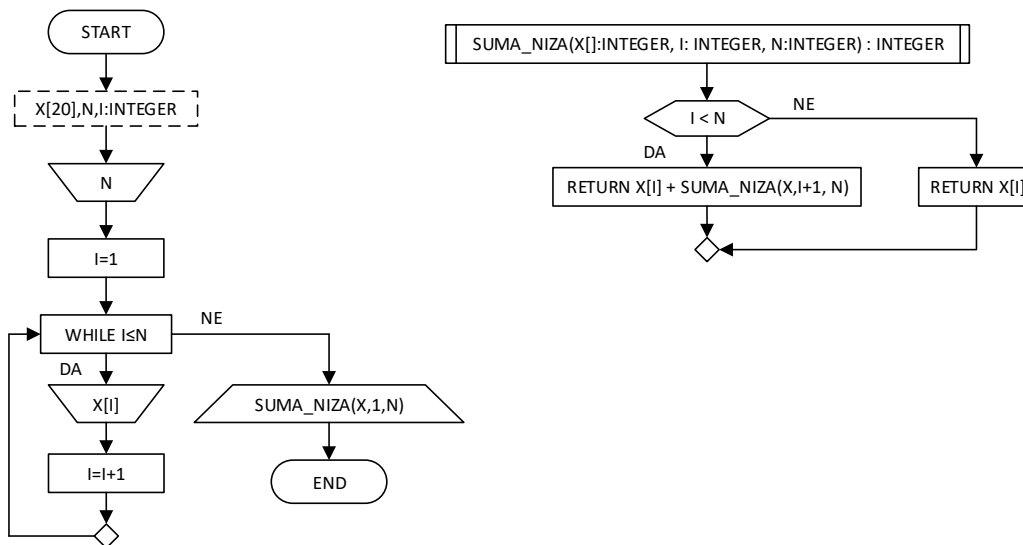
2. Kreirati algoritam kojim se učitava prirodan broj **N** i koji računa faktorijel tog broja pomoću rekurzivne funkcije **FAKT**. U glavnom programu štampati dobijenu vrijednost.



3. Kreirati algoritam kojim se učitava cijeli broj **N** i koji određuje i štampa n-ti član Fibonačijevog niza pomoću rekurzivne funkcije **FIBONACI**. U glavnom programu štampati dobijenu vrijednost.



4. Kreirati algoritam kojim se učitava niz cijelih brojeva **X**, dužine **N**, i koji računa sumu elemenata tog niza pomoću rekurzivne funkcije **SUMA_NIZA**. U glavnom programu štampati dobijenu sumu.



5. Dat je pseudokod glavnog programa i pseudokod funkcije FUN koja je realizovana rekurzivno:

A:INTEGER
A=FUN(2,6)
OUTPUT A
END

FUN(I:INTEGER,
J:INTEGER):INTEGER
IF I=0
RETURN 1
ELSEIF I>J
RETURN I+FUN(I-1,J)
ELSE
RETURN J*FUN(I,J/2)
ENDIF

Odrediti vrijednost promjenljive A nakon izvršavanja ovog koda.

A=FUN(2,6)=?

FUN(2,6) vraća 6*FUN(2,3)

FUN(2,3) vraća 3*FUN(2,1)

FUN(2,1) vraća 2+FUN(1,1)

FUN(1,1) vraća 1*FUN(1,0)

FUN(1,0) vraća 1+FUN(0,0)

FUN(0,0) vraća 1

Prema tome:

$$A = FUN(2,6) = 6 * FUN(2,3) = 6 * 3 * FUN(2,1) = 6 * 3 * (2 + FUN(1,1)) = 6 * 3 * (2 + 1 * FUN(1,0)) = 6 * 3 * (2 + 1 * (1 + FUN(0,0))) = 6 * 3 * (2 + 1 * (1 + 1)) = 72$$

Odgovor: A=72.

6. Dat je niz cijelih brojeva:

64 18 12 76 8 2 56 23

Sortirati dati niz u rastući redosljed:

- algoritmom **ponovljenih minimuma**;
- bubble sort** algoritmom;
- insertion sort** algoritmom.

Prilikom sortiranja je potrebno ispisati raspored elemenata niza nakon svake zamjene elemenata.

a) Ponovljeni minimum

```

64 18 12 76 8 2 56 23
18 64 12 76 8 2 56 23
12 64 18 76 8 2 56 23
8 64 18 76 12 2 56 23
2 64 18 76 12 8 56 23
2 18 64 76 12 8 56 23
2 12 64 76 18 8 56 23
2 8 64 76 18 12 56 23
2 8 18 76 64 12 56 23
2 8 12 76 64 18 56 23
2 8 12 64 76 18 56 23
2 8 12 18 76 64 56 23
2 8 12 18 64 76 56 23
2 8 12 18 56 76 64 23
2 8 12 18 23 76 64 56
2 8 12 18 23 64 76 56
2 8 12 18 23 56 76 64
2 8 12 18 23 56 64 76

```

b) Bubble sort

```

64 18 12 76 8 2 56 23
18 64 12 76 8 2 56 23
18 12 64 76 8 2 56 23
18 12 64 8 76 2 56 23
18 12 64 8 2 76 56 23
18 12 64 8 2 56 76 23
18 12 64 8 2 56 23 76
12 18 64 8 2 56 23 76
12 18 8 64 2 56 23 76
12 18 8 2 64 56 23 76
12 18 8 2 56 64 23 76
12 18 8 2 56 23 64 76
12 8 18 2 56 23 64 76
12 8 2 18 56 23 64 76
12 8 2 18 23 56 64 76
8 12 2 18 23 56 64 76
8 2 12 18 23 56 64 76
2 8 12 18 23 56 64 76

```

c) Insertion sort

```

64 18 12 76 8 2 56 23
18 64 12 76 8 2 56 23
18 12 64 76 8 2 56 23
12 18 64 76 8 2 56 23
12 18 64 8 76 2 56 23
12 18 8 64 76 2 56 23
12 8 18 64 76 2 56 23
8 12 18 64 76 2 56 23
8 12 18 64 2 76 56 23
8 12 18 2 64 76 56 23
8 12 2 18 64 76 56 23
8 2 12 18 64 76 56 23
2 8 12 18 64 76 56 23
2 8 12 18 64 56 76 23
2 8 12 18 56 64 76 23
2 8 12 18 56 64 23 76
2 8 12 18 56 23 64 76
2 8 12 18 23 56 64 76

```