

Vježba 6

Zadatak 1 – *Sub* procedura - izračun površine i dijagonale pravokutnika

Napišite program za izračun površine i dijagonale pravokutnika. Podjela poslova je ovakva:

- u **glavnoj** proceduri je **upis** i **provjera stranica** te **poziv procedure** ukoliko su uvjeti zadovoljeni,
- **Sub procedura** *IzracunajPovrsinuIlijagonalu* računa i **ispisuje** površinu i dijagonalu .

Procedura *IzracunajPovrsinuIlijagonalu* ima dva parametra u koja će biti prosljeđene kopije vrijednosti brojeva unesenih u *Main* proceduri.

Omogućite ponavljanje izvršavanja sve dok korisnik za vrijednosti stranica unosi pozitivne brojeve.

```
Stranica a--> 3,4
Stranica b--> 6,32
P=21,488
d=7,177

Stranica a--> 7,8
Stranica b--> 5
P=39
d=9,265

Stranica a--> 0
```

Zadatak 2 – Funkcijske procedure - izračun površine i dijagonale pravokutnika

Isti zadatak sada riješite pomoću **dvije** funkcije- *IzracunajPovrsinu* i *IzracunajDijagonalu*.

Podjela poslova sada je ovakva:

- u **glavnoj** proceduri je **upis** i **provjera stranica** te **pozivi funkcija** i **ispis rezultata**,
- Function procedure** *IzracunajPovrsinu* i *IzracunajDijagonalu* **vraćaju (Return)** pozivnom kodu **rezultate izraza** za računanje površine i dijagonale .

Kopirajte kod prethodnog zadatka i napravite potrebne promjene.

Zadatak 3 - *Function* procedure

Napišite **funkciju** *Potencija(baza, eksponen)* koja vraća vrijednost izraza $baza^{eksponen}$.

Npr: *Potencija(2,3)* = $2*2*2$.

Pretpostavite da su baza i eksponen pozitivne cjelobrojne vrijednosti različite od nule.

Napišite program koji od korisnika prihvaća vrijednost za bazu i eksponent te koristi funkciju *Potencija* za izračun. Unutar funkcije upotrijebite *For...Next* petlju za ponavljajuće izvođenje izraza unutar petlje. Broj ponavljanja treba biti jednak vrijednosti prosljeđenog argumenta *eksponent*. Funkcija **vraća vrijednost izračuna**.

Napomena: Nemojte koristiti operator "^", u ovom zadatku definirajte potenciranje pomoću množenja.

Zadatak 4 – Funkcijske procedure – prosti brojevi

Napišite program koji učitava prirodne brojeve sve dok se ne učitava prost broj. Tada se ispisuje poruka nakon koliko brojeva je učitana taj broj (npr. „Prost broj 11 učitana je nakon 6 složenih brojeva“ ili neka slična poruka) .

Ponavljajući **unos** broja i **ispis** poruke je u glavnoj proceduri. Svaki učitani broj šalje se na provjeru funkciji *ProvjeriJeLiProst* . Funkcija **vraća** *True* ako je broj prost, inače vraća *False*.

Prost broj je prirodni broj veći od 1, djeljiv jedino sa brojem 1 i samim sobom.

```
Unesite prirodni broj: 12
Unesite prirodni broj: 34
Unesite prirodni broj: 98
Unesite prirodni broj: 87
Unesite prirodni broj: 95
Unesite prirodni broj: 2
Prost broj 2 učitana je nakon 5 složenih brojeva
```

Za ponavljajući unos broja koristite **Do...Loop Until** petlju a uvjet ponavljanja je zapravo poziv funkcije. Ako funkcija vrati *False* (broj nije prost), unos će se ponoviti, ako vrati *True* (broj je prost) izlazi se iz petlje i ispisuje poruka.

Zadatak 5 - Funkcijske procedure - broj ponavljanja zadane znamenke u broju

Napišite **funkciju** koja će za zadani cijeli broj **odrediti** koliko se puta zadana znamenka pojavljuje u upisanome broju i taj podatak **vratiti** pozivnom kodu. Korisnik **unos** **broj** i traženu **znamenku** u **Main** proceduri. **Ispis** broja pojavljivanja zadane znamenke se također vrši u **Main** proceduri. Omogućite ponavljanje unosa brojeva sve dok korisnik ne unese nulu.

```
unesite broj--> 2122
unesite znamenku--> 2
    Znamenka 2 u broju 2122 pojavljuje se 3 puta

unesite broj--> 3546
unesite znamenku--> 8
    Znamenka 8 u broju 3546 pojavljuje se 0 puta

unesite broj--> 0
Press any key to continue . . .
```