1. Napisati program koji za uneseni prirodni broj n računa broj n! (n faktorijela). U program se unosi prirodni broj n preko tipkovnice, a program ispisuje n!

n!=1\*2\*3\*...\*n

Npr. za n=4, rješenje je 1\*2\*3\*4=24

Napomena. Prirodni brojevi su {1,2,3,...}

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća broj faktorijela u glavni program. n se prosljeđuje u funkciju. Deklaracija funkcije:

int broj\_faktorijela(int n);

3. U glavnom programu učitati n kao broj članova 1D integer matrice. Stvoriti 1D dinamički alociranu integer matricu, s učitanim brojem članova. Potom u svaki element polja učitati pseudoslučajni broj iz intervala 0-999 (pri svakom pokretanju programa trebaju se generirati različiti pseudoslučajni brojevi).

Potom ispisati generiranu matricu na ekran; od početka do kraja i od kraja do početka. Na kraju osloboditi memoriju.

1. Napisati program koji za učitani prirodni broj n izračuna sumu prirodnih brojeva do n (uključivo n). Unosi se n preko tipkovnice, a ispisuje se suma brojeva do n na ekran.

Npr. za n=10 rješenje je 1+2+3+4+5+6+7+8+9+10=55

Napomena. Prirodni brojevi su {1,2,3,...}

2. Riješiti program 1 pomoću funkcije, s time da se suma brojeva vraća preko adrese lokalne varijable stvorene u glavnom programu. n se također prosljeđuje u funkciju. Deklaracija funkcije:

void suma\_brojeva(int n, int\* suma);

3. Svaki član strukture neka sadrži ime i prezime kao string. U glavnom programu učitati broj elemenata strukture. Potom stvoriti dinamičko polje sa zadanim brojem elemenata strukture i učitati sve elemente. Potom ispisati sve elemente strukture čije prezime počinje sa 'b' ili 'B' i na kraju osloboditi memoriju.

1. Napisati program koji učitava prirodni broj n, a ispisuje sumu svih parnih prirodnih brojeva do (i uključivo n, za n paran) n.

Npr. za n=10, rješenje je 2+4+6+8+10=30

Napomena. Prirodni brojevi su {1,2,3,...}

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća sumu parnih brojeva u glavni program. n se prosljeđuje u funkciju. Deklaracija funkcije:

int suma\_parnih(int n);

3. U glavnom programu učitavati rečenicu po rečenicu (može sadržavati 1 ili više riječi, a završava kad korisnik pritisne "enter") kao string i upisivati u redak po redak tekstualne datoteke. Upisivanje se završava kad se za string učita "0" (koja se ne upisuje u datoteku). Nakon toga treba sve rečenice iz datoteke učitati u jedan novi string i taj string ispisati na ekran. Također treba ispisati na ekran duljinu novog stringa, te broj slova 'a' i A'.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje broj slova 'a' i 'A' na ekran. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je 2.

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća broj slova u glavni program. string se prosljeđuje u funkciju. Deklaracija funkcije:

int broj\_slova(string recenica);

3. U glavnom programu učitavati cijeli broj (integer) i upisivati u redak po redak tekstualne datoteke. Upisivanje se završava kad se za broj učita 0 (koja se ne upisuje u datoteku). Nakon toga u glavnom programu treba izračunati sumu i umnožak svih upisanih brojeva (pomoću stvorene datoteke) i napisati rezultat na ekran.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje broj razmaka na ekran. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je 1.

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća broj razmaka u glavni program. string se prosljeđuje u funkciju. Deklaracija funkcije:

int broj\_razmaka(string recenica);

3. U glavnom programu učitati n kao broj članova 1D matrice stringova . Stvoriti 1D dinamički alociranu string matricu, s učitanim brojem članova. Potom u svaki element polja učitati string (gdje string sadrži 1 riječ tj. do prvog razmaka). Sortirati matricu stringova uzlazno (gledano po ASCII tablici).

Potom ispisati sortiranu matricu stringova na ekran. Na kraju osloboditi memoriju.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje rečenicu u obrnutom redoslijedu na ekran. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je ".nad raboD".

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća rezultat tj. string u glavni program. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

string obrnuti\_ispis(string recenica);

3. Svaki član strukture neka sadrži 2 float broja koja predstavljaju katete pravokutnog trokuta. U glavnom programu učitati broj elemenata strukture. Potom stvoriti dinamičko polje sa zadanim brojem elemenata strukture i učitati sve elemente (tj. katete trokuta). Nakon toga treba na ekran ispisati duljine hipotenuza svih trokuta.

Napomena. hipotenuza se računa kao 

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje broj samoglasnika na ekran. Samoglasnici su: a, e, i, o, u, A, E, I, O, U. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je 3.

2. riješiti program 1. i pomoću funkcije, koja preko adrese lokalne varijable stvorene u glavnom programu vraća broj samoglasnika. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

void broj\_samogl(string recenica, int\* brsam);

3. U glavnom programu učitati n kao broj članova 1D matrice. Stvoriti 1D dinamički alociranu integer matricu, sa zadanim brojem članova. Potom u svaki element polja učitati integer broj preko tipkovnice. Sortirati matricu uzlazno (od najmanjeg prema najvećem broju) te je ispisati na ekran. Potom ispisati sortiranu matricu silazno (tj. od najvećeg prema najmanjem broju) i na kraju osloboditi memoriju.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje samoglasnike iz rečenice na ekran, u redoslijedu kojim su u rečenici. Samoglasnici su: a, e, i, o, u, A, E, I, O, U. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je "oaa".

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća rezultat tj. string u glavni program. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

string samogl(string recenica);

3. U glavnom programu učitati n kao broj članova 1D matrice. Stvoriti 1D dinamički alociranu float matricu, sa zadanim brojem članova. Potom u svaki element polja učitati float broj preko tipkovnice. Sortirati matricu silazno (od najvećeg prema najmanjem broju) te je ispisati na ekran i na kraju osloboditi memoriju.

1. Napisati program koji učitava prirodan broj n, a ispisuje sumu svih neparnih prirodnih brojeva do (i uključivo, ako je n neparan) n.

Npr. za n=10, rješenje je 1+3+5+7+9=25

Napomena. Prirodni brojevi su {1,2,3,...}

2. riješiti program 1. i pomoću funkcije, koja preko adrese lokalne varijable stvorene u glavnom programu vraća sumu neparnih brojeva u glavni program. n se prosljeđuje u funkciju. Deklaracija funkcije:

void suma\_neparnih(int n, int\* suma);

3. U glavnom programu učitavati rečenicu po rečenicu (može sadržavati 1 ili više riječi, a završava kad korisnik pritisne "enter") kao string i upisivati u redak po redak tekstualne datoteke. Upisivanje se završava kad se za string učita "0" (koja se ne upisuje u datoteku). Nakon toga treba ispisati na ekran duljine svih upisanih rečenica, te broj samoglasnika (a, e, i, o, u, A, E, I, O, U) u svakoj rečenici, pomoću novo stvorene datoteke.

Npr. rečenica "Dobar dan." treba kao rješenje tj. ispis imati "Duljina: 10, broj samoglasnika: 3".

1. Napisati program koji učitava prirodan broj n, a ispisuje sumu svih neparnih prirodnih brojeva do (i uključivo, ako je n neparan) n.

Npr. za n=10, rješenje je 1+3+5+7+9=25

Napomena. Prirodni brojevi su {1,2,3,...}

2. riješiti program 1. i pomoću funkcije, koja preko adrese lokalne varijable stvorene u glavnom programu vraća sumu neparnih brojeva u glavni program. n se prosljeđuje u funkciju. Deklaracija funkcije:

void suma\_neparnih(int n, int\* suma);

3. U glavnom programu učitati n kao broj članova 1D integer matrice. Stvoriti 1D dinamički alociranu integer matricu, s učitanim brojem članova. Potom u svaki element polja učitati pseudoslučajni broj iz intervala 0-999 (pri svakom pokretanju programa trebaju se generirati različiti pseudoslučajni brojevi).

Potom ispisati generiranu matricu na ekran; od početka do kraja i od kraja do početka. Na kraju osloboditi memoriju.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje samoglasnike iz rečenice na ekran, u redoslijedu kojim su u rečenici. Samoglasnici su: a, e, i, o, u, A, E, I, O, U. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je "oaa".

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća rezultat tj. string u glavni program. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

string samogl(string recenica);

3. Svaki član strukture neka sadrži ime i prezime kao string. U glavnom programu učitati broj elemenata strukture. Potom stvoriti dinamičko polje sa zadanim brojem elemenata strukture i učitati sve elemente. Potom ispisati sve elemente strukture čije prezime počinje sa 'b' ili 'B' i na kraju osloboditi memoriju.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje samoglasnike iz rečenice na ekran, u redoslijedu kojim su u rečenici. Samoglasnici su: a, e, i, o, u, A, E, I, O, U. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je "oaa".

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća rezultat tj. string u glavni program. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

string samogl(string recenica);

3. U glavnom programu učitavati rečenicu po rečenicu (može sadržavati 1 ili više riječi, a završava kad korisnik pritisne "enter") kao string i upisivati u redak po redak tekstualne datoteke. Upisivanje se završava kad se za string učita "0" (koja se ne upisuje u datoteku). Nakon toga treba sve rečenice iz datoteke učitati u jedan novi string i taj string ispisati na ekran. Također treba ispisati na ekran duljinu novog stringa, te broj slova 'a' i A'.

1. Napisati program koji učitava string (tj. rečenicu, koja može sadržavati 1 ili više riječi), a ispisuje rečenicu u obrnutom redoslijedu na ekran. String završava kad korisnik pritisne "enter".

Npr. za string "Dobar dan.", rješenje je ".nad raboD".

2. riješiti program 1. i pomoću funkcije, koja preko return naredbe vraća rezultat tj. string u glavni program. Početni string se prosljeđuje u funkciju. Deklaracija funkcije:

string obrnuti\_ispis(string recenica);

3. U glavnom programu učitavati cijeli broj (integer) i upisivati u redak po redak tekstualne datoteke. Upisivanje se završava kad se za broj učita 0 (koja se ne upisuje u datoteku). Nakon toga u glavnom programu treba izračunati sumu i umnožak svih upisanih brojeva (pomoću stvorene datoteke) i napisati rezultat na ekran.