

Osnovi programiranja I kolokvijum

10.11.2019.

I grupa

Na **Desktop**-u u direktorijumu **Rad** kreirati direktorijum **ImePrezime_BrIndeksa** i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Rešenje 1. zadatka **mora** da se nalazi u fajlu **Zadatak1.c**, rešenje 2. zadatka **mora** da se nalazi u fajlu **Zadatak2.c**. Od dva ponuđena zadatka birate **jedan** koji ćete raditi.

Napisati C program koji:

1. za učitane celobrojnu vrednost promenljive **n** računa vrednost funkcije $f(n)$ na sledeći način:

- ukoliko je uneti broj **n** pozitivan funkcija $f(n)$ dobija kao rezultat date sume:

$$f(n) = \sum_{i=1}^n \left(\binom{n}{i} + (3 + 6 + \dots + 3i) \right), \text{ gde je } \binom{n}{i} \text{ koeficijent binomne formule, } \binom{n}{i} = \frac{n!}{i!(n-i)!}.$$

- ukoliko je uneti broj **n** negativan ili jednak nuli, vrednost funkcije $f(n)$ se računa na sledeći način:

$$f(n) = \text{je broj formiran od neparnih cifara broja } |n| \text{ (za } n=-15267 \rightarrow 751)$$

Ulaz sadrži samo jedan ceo broj **n**, izlaz sadrži samo jedan ceo broj u formatu **na 10 mesta**.

NAPOMENA: Funkcije **abs/fabs, pow, kao i svoje funkcije** ne smete koristiti ni kod uslova definisanosti, ni kod računanja vrednosti.

Takođe, tekući član sume **OBAVEZNO** računati na osnovu prethodnog člana, ukoliko je to moguće. Zadatak se boduje i delimično, ukoliko se reši samo jedan način računanja funkcije $f(n)$.

Ulaz	Izlaz
3	37
10	1683
-1245116	1151
-56801	15

2. U transportni brod prenose se redom paketi zadatih masa kolicima date nosivosti (ona je veća ili jednaka masi svakog paketa). Pri tome pakete uvek prenosimo u celosti. Kada neki paket ne može stati u kolica zbog trenutnog prekoračenja nosivosti kolica, ta kolica prevozimo do broda, i počinjemo punjenje novih kolica. Napisati program koji prikazuje redom, za svaka kolica, broj paketa i ukupnu masu paketa prenetih njima.

U prvoj liniji standardnog ulaza nalazi se nosivost kolica **N** (prirodan broj $10 \leq N \leq 500$). U svakoj narednoj liniji nalazi se masa paketa **mi** (prirodan broj $1 \leq mi \leq N$) koji treba preneti. Ulaz se završava linijom u kojoj se nalazi broj **0**. Čim se napune neka kolica ispisuje se u jednoj liniji, broj paketa na kolicima i ukupna masa paketa stavljenih na njih (razdvojeni razmakom).

Ulaz	Izlaz
30	3 23
5	3 20
10	2 28
8	4 26
12	1 22
3	1 9
5	
15	
13	
5	
15	
2	
4	
22	
9	
0	

Broj poena: 1. zadatak – 13 poena, 2. zadatak – 18 poena

Vreme izrade: 75 minuta

Osnovi programiranja I kolokvijum

10.11.2019.

II grupa

Na **Desktop**-u u direktorijumu **Rad** kreirati direktorijum **ImePrezime_BrIndeksa** i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Rešenje 1. zadatka **mora** da se nalazi u fajlu **Zadatak1.c**, rešenje 2. zadatka **mora** da se nalazi u fajlu **Zadatak2.c**. Od dva ponuđena zadatka birate **jedan** koji ćete raditi.

Napisati C program koji:

3. za učitano celobrojnu vrednost promenljive **n** računa vrednost funkcije $f(n)$ na sledeći način:

- ukoliko je uneti broj **n** pozitivan funkcija $f(n)$ dobija kao rezultat date sume:

$$f(n) = \sum_{i=1}^n \left(\frac{(2n)!}{(2i)!(2n-i)!} + (2 + 4 + \dots + 2i) \right)$$

- ukoliko je uneti broj **n** negativan ili jednak nuli, vrednost funkcije $f(n)$ se računa na sledeći način:

$$f(n) = \text{je broj formiran od parnih cifara broja } |n| \text{ (za } n=-85267 \rightarrow 826)$$

Ulaz sadrži samo jedan ceo broj **n**, izlaz sadrži samo jedan broj, ili realan broj koji je rezultat sume **na deset mesta sa pet decimala**, ili predstavlja ceo broj u formatu **na 10 mesta**.

NAPOMENA: Funkcije **abs/fabs, pow, kao i svoje funkcije** ne smete koristiti ni kod uslova definisanosti, ni kod računanja vrednosti. Takođe, tekući član sume **OBAVEZNO** računati na osnovu prethodnog člana, ukoliko je to moguće. Zadatak se boduje i delimično, ukoliko se reši samo jedan način računanja funkcije $f(n)$.

Ulaz	Izlaz
3	24.41667
7	182.29126
-85267	826
-5580	80

4. Kružna meta koju koristimo u specijalnom pikadu izdvojena je u koncentrične kružne prstenove sa centrom u koordinatnom početku. Širina prstenova je ista. Napisati program koji za datu tačku određuje zonu kojoj pripada.

Sa standardnog ulaza unosi se ceo broj **n** ($1 \leq n \leq 50000$), koji predstavlja broj kružnih prstenova, zatim realan broj **d** koji predstavlja širinu prstenova. Nakon toga se unosi ceo broj **m** i zatim **m** parova koordinata tačaka (u svakom redu se nalaze dva realna broja, razdvojena sa po jednim razmakom), nakon svake unete tačke u novom redu ispisuje se ili indeks zone (broje se od nule, od najmanjeg prstena) kojoj tačka pripada ili tekst **izvan** ako je tačka izvan poslednje zone. Ako je tačka na granici dve zone, smatrati da pripada unutrašnjoj.

Primer:

Ulaz	Izlaz
4	0
3	3
5	1
2 1	Izvan
7.5 9	0
2.5 4.2	
10.8 11.2	
1.73 1.73	

Broj poena: 1. zadatak – 13 poena, 2. zadatak – 18 poena

Vreme izrade: 75 minuta