

PR I**Integralni ispit – 22.02.2025 – Grupa C**

Napomena: Obavezno pročitati dodatne instrukcije i informacije na kraju ovog dokumenta

Studenti koji polažu parcijalno rade samo 3. i 4. Zadatak

Zadatak 1:

Napisati program koji simulira bacanje 5 kockica, ali sa sljedećim pravilima:

1. Kockice imaju vrijednosti od 1 do 6.
2. Ponavljati bacanje kockica sve dok se ne dobiju tačno tri ponavljanja jedne vrijednosti i dva ponavljanja druge vrijednosti (npr. tri petice i dvije trojke). Redoslijed ispisanih vrijednosti kockica nije bitan.
3. Ispisati svako bacanje zasebno.
4. Na kraju ispisati ukupan broj bacanja, kao i posljednje izvučene vrijednosti kockica.

Primjer korištenja:

```
Bacanje 10: 6 1 2 6 6
Bacanje 11: 1 2 5 1 6
Bacanje 12: 3 1 4 4 4
Bacanje 13: 3 3 4 6 6
Bacanje 14: 4 2 3 6 6
Bacanje 15: 6 1 3 1 3
Bacanje 16: 2 1 2 3 6
Bacanje 17: 3 3 1 1 2
Bacanje 18: 5 3 6 2 2
Bacanje 19: 2 1 4 2 2
Bacanje 20: 5 6 6 2 1
Bacanje 21: 4 4 2 2 1
Bacanje 22: 5 4 1 5 4
Bacanje 23: 5 1 6 5 1
Bacanje 24: 1 1 1 4 3
Bacanje 25: 6 5 2 1 6
Bacanje 26: 6 1 1 6 2
Bacanje 27: 1 2 1 3 2
Bacanje 28: 4 4 6 5 6
Bacanje 29: 3 5 2 1 5
Bacanje 30: 2 4 1 6 3
Bacanje 31: 5 6 1 2 1
Bacanje 32: 6 3 2 2 6
Bacanje 33: 6 4 4 3 1
Bacanje 34: 6 5 1 5 6
Bacanje 35: 1 4 1 2 6
Bacanje 36: 5 2 3 3 1
Bacanje 37: 5 5 3 4 4
Bacanje 38: 6 1 1 1 4
Bacanje 39: 4 1 3 2 1
Bacanje 40: 4 6 5 5 6
Bacanje 41: 2 2 1 1 1
Ukupan broj bacanja: 41
Posljednje izvucene vrijednosti: 2 2 1 1 1
```

Zadatak 2:

Napisati program koji omogućuje korisniku unos jedne od 3 opcije:

- 1 - Odabir ove opcije primorava korisnika na unos dvije cjelobrojne vrijednosti **m** i **n**. Program treba ispisati sve parne brojeve u intervalu **m** i **n**.
- 2 - Odabir ove opcije primorava korisnika na unos dvije cjelobrojne varijable **m** i **n**. Program treba da ispiše sve proste brojeve koji se nalaze između **m** i **n**.
- 3 - Odabir ove opcije označava izlaz iz programa.

Ukoliko korisnik odabere opciju koja nije niti jedna od tri navedene, program se mora nastaviti izvršavati.

Primjer korištenja:

```
Odaberite opciju:
1 - Ispis parnih brojeva u intervalu [m, n]
2 - Ispis prostih brojeva u intervalu [m, n]
3 - Izlaz iz programa
Unos opcije: 1
Unesite dva cijela broj m i n: 10 50
10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
Odaberite opciju:
1 - Ispis parnih brojeva u intervalu [m, n]
2 - Ispis prostih brojeva u intervalu [m, n]
3 - Izlaz iz programa
Unos opcije: 2
Unesite dva cijela broj m i n: 10 50
11 13 17 19 23 29 31 37 41 43 47
Odaberite opciju:
1 - Ispis parnih brojeva u intervalu [m, n]
2 - Ispis prostih brojeva u intervalu [m, n]
3 - Izlaz iz programa
Unos opcije: 3
Izlazak iz programa...
```

20 bodova

Zadatak 3:

Napišite program u kojem je potrebno unijeti veličinu dinamičkog niza cjelobrojnih vrijednosti. Nakon unosa veličine niza, program treba da uradi dinamičku alokaciju niza. Zatim je potrebno da se elementi niza popune sa slučajnim vrijednostima u opsegu od 0 do 99999 [uključujući i granične vrijednosti] **i to isključivo pomoću rekurzivne funkcije**. Zatim, definirati **rekurzivnu funkciju** za prebrojavanje broja cifara u broju, te zatim pronaći (**nerekurzivno**) broj sa najvećim brojem cifara i broj sa najmanjim brojem cifara te ispisati ih oba. Ukoliko više brojeva ima isti broj cifara, vratiti onaj broj koji je prvi po redoslijedu.

Obavezno uraditi sve neophodne dealokacije.

U zadatku obavezno koristiti operator '*' umjesto '[]' prilikom dereferenciranja adresa/pokazivača. Također, ne smije se koristiti logaritamska funkcija za prebrojavanje broja cifara.

Primjer korištenja:

```
Unesite veličinu niza: 20
Generisani niz: 10775 10348 23021 2329 5372 14292 950 11003 23516 16093 14300 7684 4754 12425 20909 29530 23673 24154 27118 30906
Broj sa najviše cifara: 10775
Broj sa najmanje cifara: 950
```

20 bodova

Zadatak 4:

Neka je dat sljedeći prototip strukture Glumac:

```
struct Glumac {
    char* ime = nullptr;
    char* prezime = nullptr;
    char drzava[40] = "";
    int brojFilmova = 0; // označava broj redova u dinamičkoj matrici
    int brojKorisnika = 0; // označava broj kolonu u dinamičkoj matrici
    int** ocjene = nullptr; // matrica dimenzija (brojFilmova × brojKorisnika)

    void unos(); // funkcija članica za alokaciju i unos vrijednosti
    float* getAveragesByUsers();
    void ispis();
    void dealokacije();
};
```

Uraditi implementaciju zadanih funkcija članica i po potrebi definirati i implementirati pomoćne globalne funkcije. Zatim, u main funkciji dodati sljedeće stavke:

- Kreiranje dinamičkog objekta tipa *Glumac*
- Poziv funkcije članice unos:
 - **Napomena: Obavezno postaviti vrijednosti matrice „ocjene“ na slučajne vrijednosti [opseg od 1 do 10]. Ocjene se ne unose ručno.**
 - Sve ostale atribute [ime, prezime, ...] je potrebno ručno unijeti preko tastature.
- Poziv funkcije članice ispis
- Dohvaćanje niza prosječnih ocjena glumca za svakog od korisnika (**koristiti dinamičku memoriju**).
- Poziv funkcije za dealokaciju objekta.
- Dealokacija niza prosječnih ocjena.

Obavezno uraditi sve neophodne dealokacije.

U zadatku obavezno koristiti operator '*' umjesto '[]' prilikom dereferenciranja adresa/pokazivača.

```
Unesite ime:
Ryan
Unesite prezime:
Gosling
Unesite drzavu:
Kanada
Unesite broj filmova:
13
Unesite broj korisnika:
4
Ime i prezime: Ryan Gosling
Drzava: Kanada
Broj filmova: 13
Broj korisnika: 4
-----
  2   8   5   1
10   5   9   9
  3   5   6   6
  2   8   2   2
  6   3   8   7
  2   5   3   4
  3   3   2   7
  9   6   8   7
  2   9  10   3
  8  10   6   5
  4   2   3   4
  4   5   2   2
  4   9   8   5
-----
Ispis prosjecnih ocjena glumca po svakom korisniku:
4.53846   6 5.53846 4.76923
```

40 bodova

Kratke upute za spašavanje rješenja na ispitu [korištenje FTP servera]

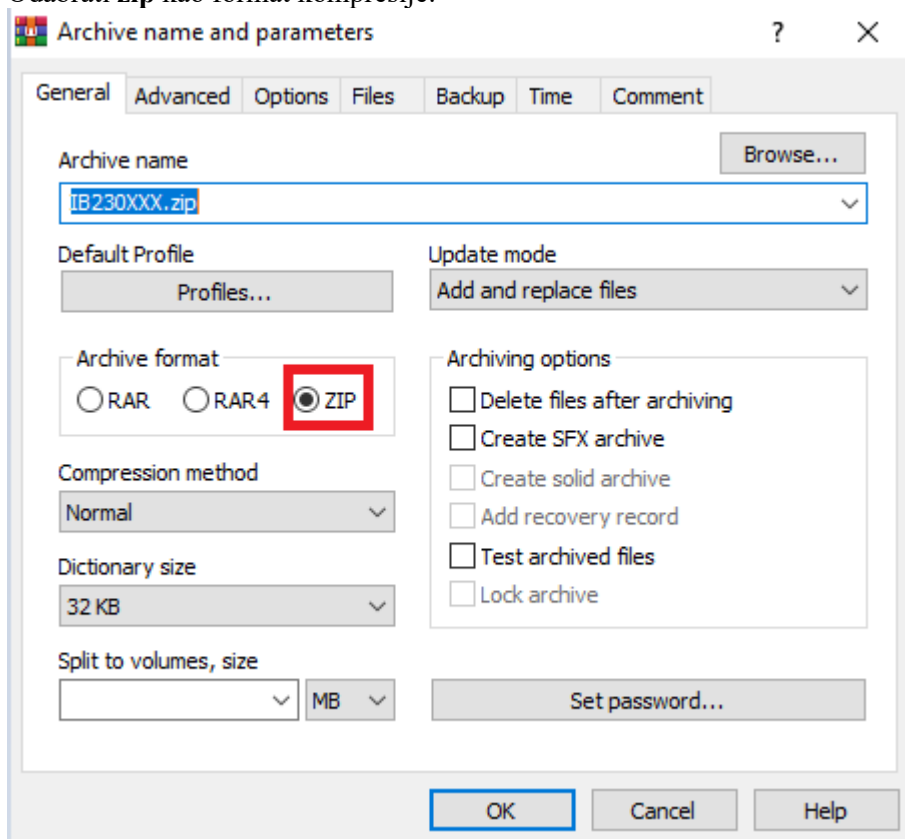
1. Rješenja zadataka imenovati na sljedeći način:

Name	Status	Date modified	Type
*+ Zadatak 1.cpp	✓	31. 1. 2024. 13:49	C++ Source
*+ Zadatak 2.cpp	✓	31. 1. 2024. 13:49	C++ Source
*+ Zadatak 3.cpp	✓	31. 1. 2024. 13:49	C++ Source
*+ Zadatak 4.cpp	✓	31. 1. 2024. 13:49	C++ Source

2. Dodati sve *.cpp fajlove u jedan folder sa nazivom **IB240***** (broj indeksa studenta)

3. Uraditi kompresiju

- Desni klik na folder i odabrati opciju *Add to archive*
- Odabrati **zip** kao format kompresije:



Rezultat kompresije je fajl sa nazivom **IB230***.zip**

4. Priključiti se na ftp server (<ftp.fit.ba>) preko File Explorer-a sa sljedećim login podacima:

- Username: **student_km**
- Password: **student_km**

5. Navigirati do sekcije za upload fajla: folder **Upload** → podfolder **PRI** → podfolder **Grupa A**

6. Kopirati fajl **IB240***.zip** u odgovarajući folder

Zadaci se boduju djelimično ako je bar jedan dio traženih funkcionalnosti ostvaren. Zakomentirani dijelovi koda neće biti pregledani. **Zabranjeno je postavljati cijeli VS Solution ili Project.** U slučaju postavljanja cijelog solutiona ili projekta zadatak neće biti pregledan.