

Objektno orijentirano programiranje

Zadaća 2

Tuzla, novembar 2022.

Zadatak 1

Napisati program koji predstavlja konverter mjernih jedinica.

Program treba da sadrži:

1. Konverter mjernih jedinica temperature. Omogućiti unos temperature u Kelvinima, stepenima Celzijusa ili stepenima Fahrenheita.
2. Konverter mjernih jedinica brzine. Dati izbor unosa brzine u miljama ili kilometrima po satu. Ispisati brzinu u drugoj mjernoj jedinici.
3. Konverter dužine između svjetlosnih godina i kilometara, stopa i metara te inča i centimetara.
4. Konverter mase između kilograma i funte (kg ↔ lb)
5. Konverter potrošnje goriva između litara na 100 kilometara u milje po galonu.

Program kada se starta treba da ispiše meni za korisnika. Primjer korištenja programa je dat ispod.

```
Welcome to Unit converter. Please enter one of the following options:
1. Temperature
2. Speed
3. Length
4. Weight
5. Fuel economy
Your choice: 1
Please choose converter:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
3. Celsius to Kelvin
4. Kelvin to Celsius
5. Fahrenheit to Kelvin
6. Kelvin to Fahrenheit
Your choice: 2
Enter temperature in degrees Fahrenheits: 105
105 degrees Fahrenheit is 40.5556 degrees Celsius.
```

Obratiti pažnju na organizaciju programa. Obavezno konstante za različite kategorije smjestiti u različite namespace-e.

Na primjer:

```
namespace Temperature {
const double celsiusKelvinCoeff = 273.15;
}
namespace Length {
const double inchCmCoeff = 2.54;
}
```

Note: Obratiti pažnju na nevalidne unose. Negativna vrijednost Kelvina ne postoji, temperatura ispod -273.15 Celzija također.

Svjetlosna godina je preko deset magnituda veća od kilometra, obratiti pažnju da se ne desi overflow pri računanju.

Zadatak 2

Napisati program koji će simulirati 16-bitni registar koristeći primitivne tipove iz C++-a. Potrebno je implementirati operacije setovanja i resetovanja bita na registru. Dio koda je već ispisan (u prilogu zadatke nalazi se file: zadatak2.cpp), na Vama je da dopunite dijelove koji nedostaju mijenjajući priložen file po želji. Jedan use-case navedenog programa je dat ispod.

```
1. Print register
2. Set bit in register
3. Reset bit in register
4. Exit
Enter option: 1
0000000000000000
```

```
1. Print register
2. Set bit in register
3. Reset bit in register
4. Exit
Enter option: 2
Enter bit number: 0
New register value: 1
0000000000000001
```

```
1. Print register
2. Set bit in register
3. Reset bit in register
4. Exit
Enter option: 2
Enter bit number: 3
New register value: 9
0000000000001001
```

```
1. Print register
2. Set bit in register
3. Reset bit in register
4. Exit
Enter option: 3
Enter bit number: 0
New register value: 8
0000000000001000
```

Nakon ovoga, proširiti zadatak za još jedan registar te prije biranja operacije nad registrom odabrati koji registar korisnik želi modifikovati (ispisati). Osim toga, uvesti još jednu opciju

gdje se vrijednosti dva registra zamjene (registar a dobije vrijednost registra b, te registar b dobije vrijednost registra a).

Zadatak 3

Napisati program koji za unesena tri cijela broja utvrđuje da li predstavljaju dužine stranica pravouglog trougla. Program potom modificirati tako da za korisnički unesen cijeli broj određuje sve moguće pravougule trougle čije su dužine stranica takođe cijeli brojevi te su manje ili jednake unesenom broju. Primjerice, za uneseni broj 20, ispis treba biti:

```
(3, 4, 5)
(6, 8, 10)
(5, 12, 13)
(9, 12, 15)
(8, 15, 17)
(12, 16, 20)
```

Zadatak 4

Napisati program koji prima dvije riječi sa standardnog ulaza te provjerava [Hamming-ovu udaljenost](#) između unesena dva string-a. Pri provjeri udaljenosti ne obraćati pažnju na mala i velika slova (npr. Ai a su identični karakteri). Primjer izvršenja programa na [sljedećem linku](#).

Zadatak 5

Napisati program koji od korisnika očekuje unos cijelog broja, a potom u vektor sprema pojedinačne cifre istog. Koristeći kreirani vektor, ispisati sve cifre broja u obrnutom redoslijedu, najmanju i najveću cifru te sumu svih cifara i njihov proizvod.

Na primjer, za unos: 1234567, program treba ispisati:

```
7654321
1
7
28
5040
```

Zadatak 6

Napisati program koji će simulirati rad kalkulatora, izvršavajući operacije +, -, *, /, %, ^.
Program od korisnika očekuje unos dva cijela broja i karakter koji definira operaciju.
Potrebno je vršiti validaciju unosa (na primjer, ne dozvoliti dijeljenje sa nulom).
Primjer korištenja programa, sa različitim unosom, kao i očekivani ispis su prikazani ispod.

Ulaz	Izlaz
1 + 2	3
10 / 3	3.333
2 % 3	2
1 / 0	Dijeljenje sa nulom je nedefinirana operacija.
10 * a	Nevalidan unos: 'a' nije broj.
c + b	Nevalidan unos: 'c' i 'b' nisu brojevi.
2 ^ 3	8

Zadatak 7

Napisati program koji ima identičan rezultat kao što je prikazano na narednoj slici:

```
Unesi rijec: rijec1
Unesi rijec: rijec1111
Unesi rijec: a
Unesi rijec: ab
Unesi rijec: abcd
Unesi rijec: jkdfdhfjk
Unesi rijec: dslkfjlkkl dsfkl
Unesi rijec: lkdsjfklds j
Unesi rijec: j
Unesi rijec: dflksjflkdsjflkdsfjsdklfjk
Unesi rijec: dkskdlfsl dkfjdsklfjdsklfj
Unesi rijec: jfd
Unesi rijec:
+++++
+
+ a
+ ab
+ abcd
+ dflksjflkdsjflkdsfjsdklfjk
+ dkskdlfsl dkfjdsklfjdsklfj
+ dslkfjlkkl dsfkl
+ j
+ jfd
+ jkdfdhfjk
+ lkdsjfklds j
+ rijec1
+ rijec1111
+
+++++
```

Napomena: Obratiti paznju da su rijeci ispisane sortirane leksikografski.

Zadatak 8

Napisati program koji omogućava izvlačenje brojeva lota. Program od korisnika traži da unese ukupan broj kuglica koje učestvuju u izvlačenju i broj kuglica koje će biti izvučene. Za generiranje nasumičnih brojeva u opsegu **[0-RAND_MAX]** koristiti funkciju **rand()** iz zaglavlja **<cstdlib>**.

Primjer generiranja nasumičnih brojeva:

```
#include <cstdlib>
```

```
#include <iostream>

int main() {
    srand(time(NULL)); // Omogućava da se svaki put prilikom pokretanja
    programa dobije novi random broj
    int a = rand(); // Generira random cijeli broj.
    int b = rand() % 100; // Generira random broj u opsegu od 0-100
    int c = 25 + rand() % 50; // Generira random broj u opsegu 25-75
    std::cout << "a = " << a << std::endl;
    std::cout << "b = " << b << std::endl;
    std::cout << "c = " << c << std::endl;
}
```

Program treba da ispise brojeve u sekvenci kako su izvuceni i u sortiranoj sekvenci.
 Primjer rada programa:

```
Unesite ukupan broj kuglica: 36
Unesite broj kuglica za izvlacenje: 6
```

Izvuceno	Sortirano
33	4
24	11
26	23
11	24
4	26
23	33

Dodatno: Sortirati izvučene kuglice od veće ka manjoj, kao i na način da sve parne kuglice budu na početku a sve neparne na kraju.

Zadatak 9

Napisati program koji omogućava unos proizvoljnog broja riječi sa ulaza.
 Program nakon toga ispisuje unesenu listu riječi, ukupan broj karaktera u svim riječima i najdužu unesenu riječ koja se nalazi unutar okvira čije su ivice sačinjene od karaktera '+'.
 Primjer izvršavanja programa je prikazan na narednoj slici:

```
Unesi rijec: cat
Unesi rijec: hex
Unesi rijec: dog
Unesi rijec: door
Unesi rijec: bottle
Unesi rijec: glass
Unesi rijec: cold
Unesi rijec: television
Unesi rijec: computer
Unesi rijec:
=====
Lista rijeci
cat hex dog door bottle glass cold television computer
Ukupan broj karaktera: 46
=====
Najduza rijec
+++++
+           +
+ television +
+           +
+++++
```