

Vježbe 4.

Zadaci

1) Kreirati program koji će nasumično odabrati i u konzolnom prozoru ispisati:

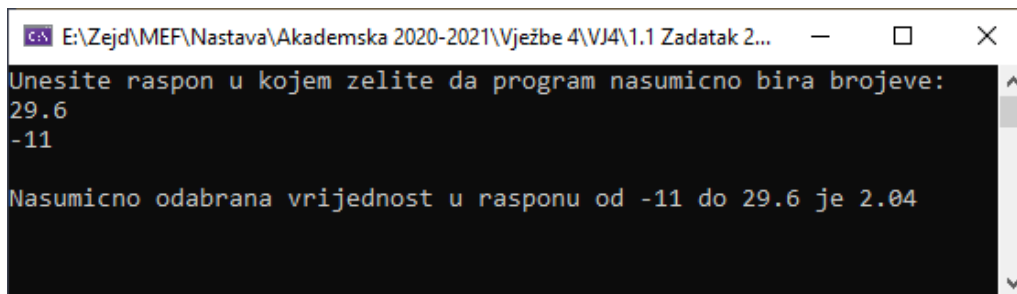
- jedan realan broj u opsegu od 0.0 do 1.0,
- jedan cijeli broj u opsegu od 50 do 90,
- i jedan realan broj u opsegu od 50.0 do 90.0

2) Kreirati program koji će od korisnika zahtijevati da unese dva realna broja koja predstavljaju granice raspona u kojem će program nasumično birati vrijednosti. Jednu realnu vrijednost program će odabrati i u konzolnom prozoru ispisati tu vrijednost zaokružen na dvije decimale.

Napomena:

- Pretpostaviti da će korisnik unijeti dva realna broja ali nije poznato koju će vrijednost unijeti kao gornju, a koju kao donju granicu.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\1.1 Zadatak 2...
Unesite raspon u kojem zelite da program nasumicno bira brojeve:
29.6
-11

Nasumicno odabrana vrijednost u rasponu od -11 do 29.6 je 2.04
```

3) Kreirati program koji će u konzolnom prozoru ispisati prvih 10 pozitivnih cijelih brojeva jedan ispod drugog.

Napomena:

- Koristiti for, while i do - while iteracionu strukturu pri definisanju programa.

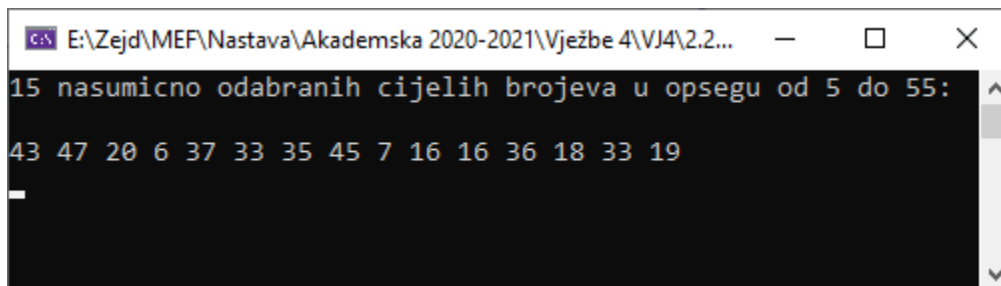
4) Kreirati program koji će nasumično odabrati i u konzolnom prozoru ispisati 15 cijelih brojeva u rasponu od 5 do 55 sa razmaknicom između svakog broja.

Napomena:

- Koristiti for, while i do - while iteracionu strukturu pri definisanju programa.

- Potrebno je uključiti obje granice u kojem se nasumično odabiru brojevi - ova napomena će uvijek važiti kad se u zadatku navede neki raspon (od _ do _) da se podrazumjeva uključivanje obje granice intervala, ako se drugačije ne navede

Ispis u konzolnom prozoru nakon izvršavanja programa:



5) Napisati program koji će ispisati sve cijele brojeve u rasponu od 0 do 100 sa korakom 5. Brojeve treba ispisati u konzolnom prozoru sa jednom razmaknicom između svakog broja.

Napomena:

- Koristiti for, while i do - while iteracionu strukturu pri definisanju programa.

6) Napisati program koji će ispisati u konzolnom prozoru sve cijele brojeve u intervalu od 0 do 100 sa korakom kojeg korisnik određuje upisujući njegovu vrijednost u konzolnom prozoru.

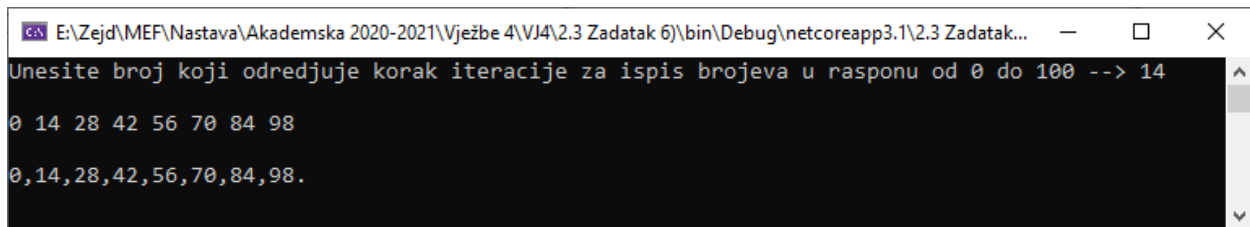
* Potrebno je ispisati sve brojeve u jednom redu odvojene razmaknicom.

** Potrebno je ispisati sve brojeve, odvojene zarezom (nakon zadnjeg broja program ispisuje tačku) i brojeve treba ispisati tako da su odvojeni i sa gornje i donje strane konzolnog prozora jednim praznim redom.

Napomena:

- Pretpostaviti da će korisnik unijeti validnu vrijednost koraka tj. unijet će pozitivan cijeli broj manji ili jednak broju 100.

Ispis u konzolnom prozoru nakon izvršavanja programa (* i **):

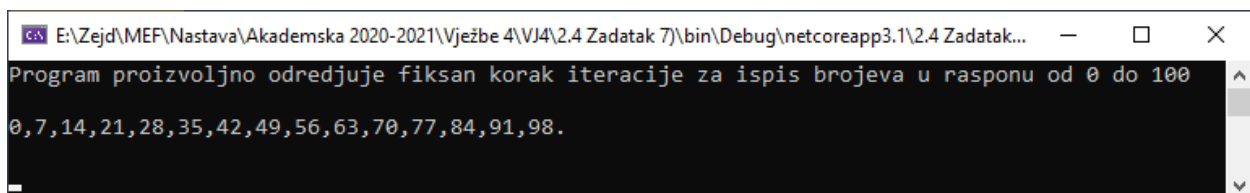


```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vjezbe 4\VJ4\2.3 Zadatak 6)\bin\Debug\netcoreapp3.1\2.3 Zadatak...  
Unesite broj koji odredjuje korak iteracije za ispis brojeva u rasponu od 0 do 100 --> 14  
0 14 28 42 56 70 84 98  
0,14,28,42,56,70,84,98.
```

7) Kreirati konzolnu aplikaciju u kojoj će se ispisati brojevi od 0 do 100, a korak iteracije program određuje proizvoljno koristeći se objektom klasom `Random`. Program bira proizvoljan korak iteracije koji bi se trebao nalaziti u intervalu od 5 do 18 i koji je fiksni tokom izvršavanja programa.

Potrebno je ispisati sve brojeve, odvojene zarezom (nakon zadnjeg broja program ispisuje tačku) i brojeve treba ispisati tako da su odvojeni i sa gornje i donje strane konzolnog prozora jednim praznim redom (kao u primjeru).

Ispis u konzolnom prozoru nakon izvršavanja programa:

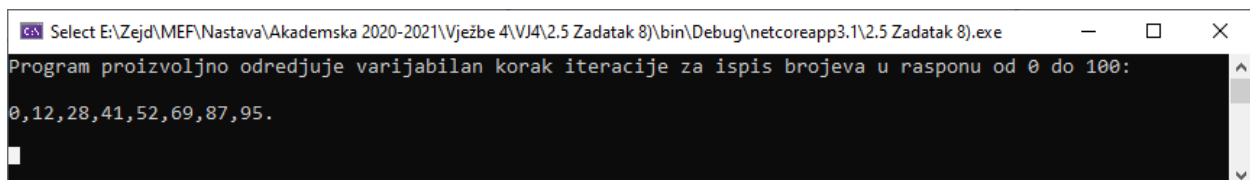


```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vjezbe 4\VJ4\2.4 Zadatak 7)\bin\Debug\netcoreapp3.1\2.4 Zadatak...  
Program proizvoljno odredjuje fiksni korak iteracije za ispis brojeva u rasponu od 0 do 100  
0,7,14,21,28,35,42,49,56,63,70,77,84,91,98.
```

8) Kreirati konzolnu aplikaciju u kojoj će se ispisati brojevi od 0 do 100, a korak iteracije program određuje proizvoljno koristeći se klasom `Random`. Program bira proizvoljan korak iteracije koji bi se trebao nalaziti u intervalu od 5 do 18 i koji se mijenja tokom izvršavanja petlje u svakoj novoj iteraciji program bira novu vrijednost koraka u intervalu od 5 do 18.

Potrebno je ispisati sve brojeve, odvojene zarezom (nakon zadnjeg broja program ispisuje tačku) i brojeve treba ispisati tako da su odvojeni i sa gornje i donje strane konzolnog prozora jednim praznim redom.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
Select E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vjezbe 4\VJ4\2.5 Zadatak 8)\bin\Debug\netcoreapp3.1\2.5 Zadatak 8).exe  
Program proizvoljno odredjuje varijabilan korak iteracije za ispis brojeva u rasponu od 0 do 100:  
0,12,28,41,52,69,87,95.
```

9) Prethodna tri zadatka objediniti i dozvoliti korisniku da bira jednu od tri opcije:

- 1) Korak iteracije određuje korisnik. Za određivanje koraka i ispis brojeva u konzolnom prozoru koristiti korisnički kreiranu metodu `IspisBrojevaKorisnik()`,
- 2) Korak iteracije određuje program u rasponu od 5 do 18 i on treba da bude fiksni tokom izvršavanja iteracija. Za određivanje koraka i ispis brojeva u konzolnom prozoru koristiti korisnički kreiranu metodu `IspisBrojevaFiksniKoraka()`,
- 3) Korak iteracije određuje program u rasponu od 5 do 18 i da se za svaku novu iteraciju taj korak mijenja. Za određivanje koraka i ispis brojeva u konzolnom prozoru koristiti korisnički kreiranu metodu `IspisBrojevaPromjenljiviKorak()`.

Program će realizovati jednu od navedenih opcija u zavisnosti od izbora, kojeg korisnik određuje unoseći vrijednost sa tastature u vidu broja 1, 2 ili 3 u konzolnom prozoru.

Potrebno je ispisati sve brojeve, odvojene jednom razmaknicom i brojeve treba ispisati tako da su odvojeni i sa gornje i donje strane konzolnog prozora jednim praznim redom.

Ispis u konzolnom prozoru nakon izvršavanja programa:

```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.6 Zadata...
Program ispisuje cijele brojeve od 0 do 100.

Izaberite jednu od tri opcije:
1) Korisnik unosi korak
2) Program bira fiksni korak
3) Program bira varijabilan korak

Vas izbor je: 3

0 11 27 35 49 67 85
```

```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.6 Zadata...
Program ispisuje cijele brojeve od 0 do 100.

Izaberite jednu od tri opcije:
1) Korisnik unosi korak
2) Program bira fiksni korak
3) Program bira varijabilan korak

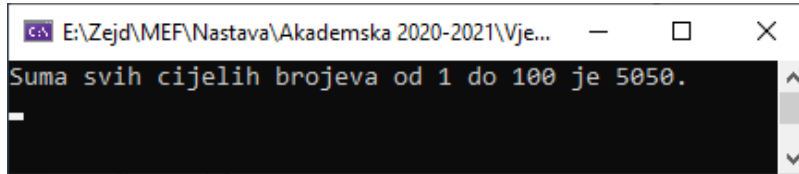
Vas izbor je: 1

Unesite vrijednost koraka iteracije: 15

0 15 30 45 60 75 90
```

10) Kreirati program u kojem će se sumirati svi pozitivni cijeli brojevi od 1 do 100, te će se njihova suma ispisati u konzolnom prozoru

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
cs E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vje...
Suma svih cijelih brojeva od 1 do 100 je 5050.
```

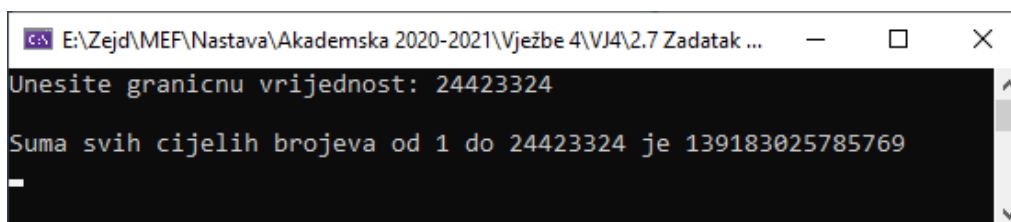
11) Prirodni brojevi do broja 10 (ne uključujući vrijednost 10) koji su djeljivi sa 3 ili sa 5 su: 3,5,6 i 9. Suma ovih brojeva je 23. Potrebno je kreirati program koji će izračunati sumu svih prirodnih brojeva do broja 1000 (ne uključujući vrijednost 10), koji su djeljivi sa 3 ili 5, i ispisati sumu u konzolnom prozoru.

12) Potrebno je kreirati program koji će izračunati sumu svih prirodnih brojeva djeljivih sa 3 ili 5. Program provjerava sve brojeve od 1 do broja kojeg korisnik unosi sa tastature (ne uključujući tu vrijednost).

Napomena:

- Pretpostaviti da će korisnik unijeti pozitivan cijeli broj
- Zbog moguće velike vrijednosti pri računanju sume koristite **long** tip podataka.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
cs E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VI4\2.7 Zadatak ...
Unesite granicnu vrijednost: 24423324
Suma svih cijelih brojeva od 1 do 24423324 je 139183025785769
```

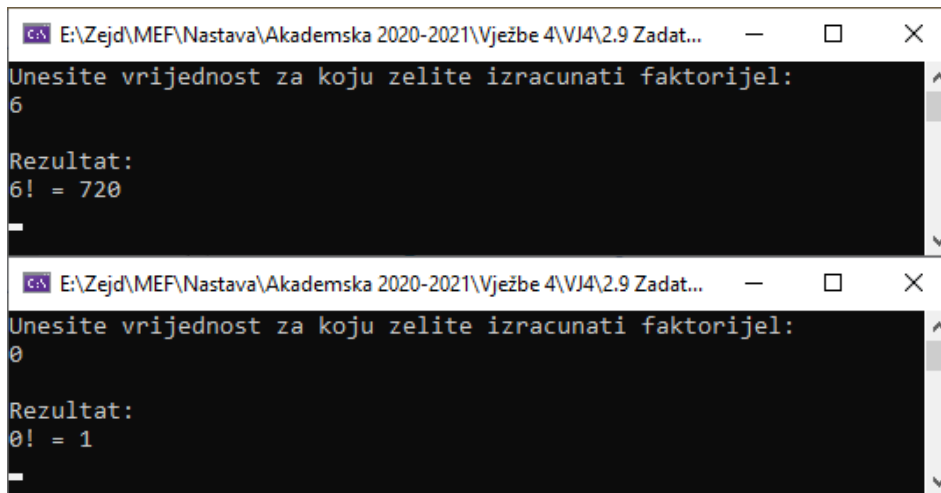
13) Kreirati program koji će računati faktorijel broja kojeg unosi korisnik sa tastature. Ako korisnik unese npr. broj 6, program će izračunati faktorijel broja $6! = (1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6) = 720$ i ispisati rezultat u konzolnom prozoru.

Napomena:

- Pretpostaviti da će korisnik unijeti pozitivan cijeli broj

- Zbog moguće velike vrijednosti pri računanju faktoriijela, koristite **long** tip podataka.

Ispis u konzolnom prozoru nakon izvršavanja programa:



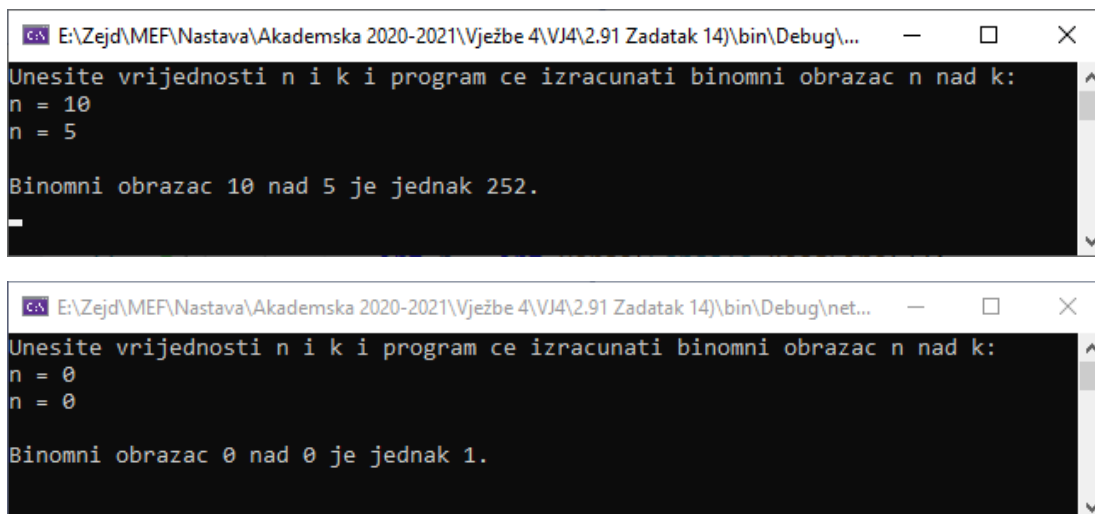
The image shows two screenshots of a Windows console window. The title bar of the window is "E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.9 Zadat...". The first screenshot shows the prompt "Unesite vrijednost za koju zelite izracunati faktorijel:" followed by the input "6". The output is "Rezultat: 6! = 720". The second screenshot shows the same prompt followed by the input "0". The output is "Rezultat: 0! = 1".

14) Binomni obrazac $\binom{n}{k}$ se računa kao $\frac{n!}{k! \cdot (n-k)!}$. Kreirati program koji će izračunati binomni obrazac za dva broja koje korisnik unosi sa tastature. Potrebno je definisati korisnički kreiranu metodu Faktorijel(), pomoću koje će se izračunavati svaki od faktoriijela unutar binomnog obrasca zasebno.

Napomena:

- Pretpostaviti da će korisnik unijeti validne pozitivne cijele brojeve i da će uslov $0 \leq k \leq n$, koji mora biti ispunjen da bi se mogao izračunati binomni obrazac, biti ispunjen za unesene brojeve sa tastaure n i k .

Ispis u konzolnom prozoru nakon izvršavanja programa:



The image shows two screenshots of a Windows console window. The title bar of the window is "E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.91 Zadatak 14)\bin\Debug\...". The first screenshot shows the prompt "Unesite vrijednosti n i k i program ce izracunati binomni obrazac n nad k:" followed by the inputs "n = 10" and "n = 5". The output is "Binomni obrazac 10 nad 5 je jednak 252.". The second screenshot shows the same prompt followed by the inputs "n = 0" and "n = 0". The output is "Binomni obrazac 0 nad 0 je jednak 1.".

15) Prethodni zadatak riješiti koristeći kreiranu metodu `BinomniObrazac()`. Razmislite koja bi najbolja opcija bila za ovu metodu vezano za ulazne i izlazni parametar, da li će biti sa njima ili bez njih (tj. bez ulaznih parametara i da metoda ne vraća ništa `void` metoda). Kreirati metodu tako da ona bude što općenitija (pogledati klasu `Math` na koji način su definisane metode `Sin()`, `Cos()`, `Exp()` itd.)

16) Kreirati program u kojem će se od korisnika zahtijevati da unese jedan pozitivni cijeli broj i program će sabrati, pomnožiti i naći srednju vrijednost svih cijelih brojeva od 1 pa do vrijednosti koju je korisnik unio. Zadatak uraditi na tri različita načina:

1. U `Main()` izvršnoj metodi sabrati, pomnožiti i naći srednju vrijednost koristeći se iteracionim strukturama, te ispisati u konzolnom prozoru rezultat nakon izvršenih operacija

2. Za sabiranje kreirati metodu `Sabiranje1doN()` metodu, za množenje kreirati metodu `Množenje1doN()`, a za traženje srednje vrijednosti koristiti metodu `()`. Ove metode vraćaju numeričke vrijednosti kao izlazne parametre (prve dvije sigurno će vratiti cjelobrojnu vrijednost, dok druga će vratiti realnu vrijednost, ali mogu se sve tri metode definisati tako da vraćaju realne vrijednosti). Nakon toga ispisati u konzolnom prozoru vrijednosti nakon sumiranja, množenja i traženja srednje vrijednosti.

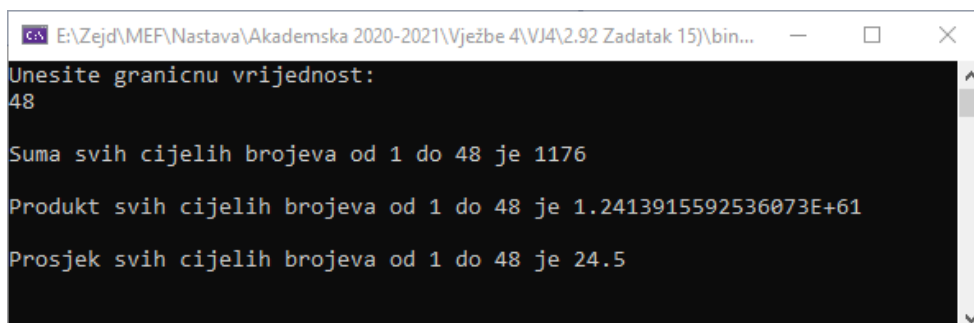
3. Za sabiranje i ispis u konzolnom prozoru kreirati metodu `Sabiranje1doNIspis()` metodu, za množenje i ispis u konzolnom prozoru kreirati metodu `Množenje1doNIspis()`, a za traženje srednje vrijednosti i ispis u konzolnom prozoru koristiti metodu `SrednjaVrijednost1doNIspis()`.

****** Ispisati prvu poruku u plavoj boji, drugu u crvenoj i trecu u zelenoj boji.

Napomena:

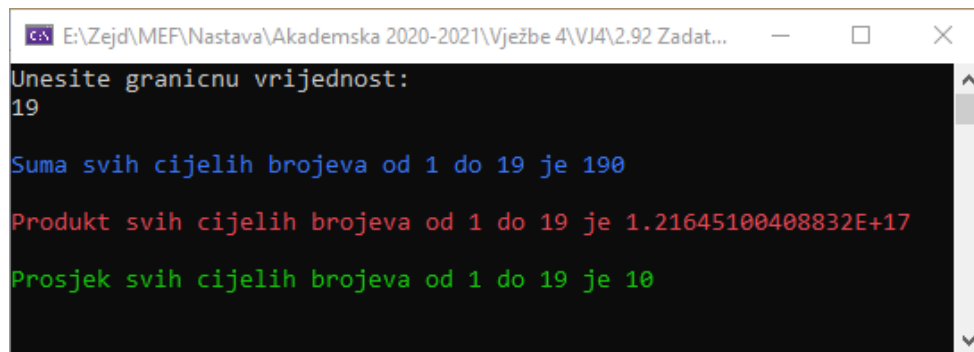
- Pretpostaviti da će korisnik unijeti pozitivan cijeli broj

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademika 2020-2021\Vježbe 4\VJ4\2.92 Zadatak 15)\bin...
Unesite granicnu vrijednost:
48
Suma svih cijelih brojeva od 1 do 48 je 1176
Produkt svih cijelih brojeva od 1 do 48 je 1.2413915592536073E+61
Prosjeck svih cijelih brojeva od 1 do 48 je 24.5
```

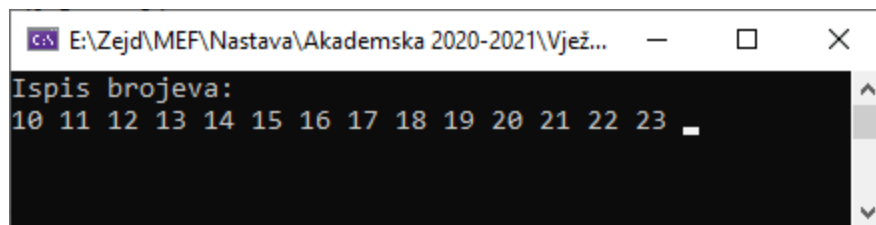
**



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.92 Zadat...
Unesite granicnu vrijednost:
19
Suma svih cijelih brojeva od 1 do 19 je 190
Produkt svih cijelih brojeva od 1 do 19 je 1.21645100408832E+17
Prosjeck svih cijelih brojeva od 1 do 19 je 10
```

17) Kreirati program u kojem će se ispisati u konzolnom prozoru svi pozitivni cijeli brojevi od 10 do 75 odmaknuti sa jednom razmaknicom. Ukoliko je jedan od brojeva iz ovog intervala djeljiv sa 8 i sa 3 prekinuti ispis.

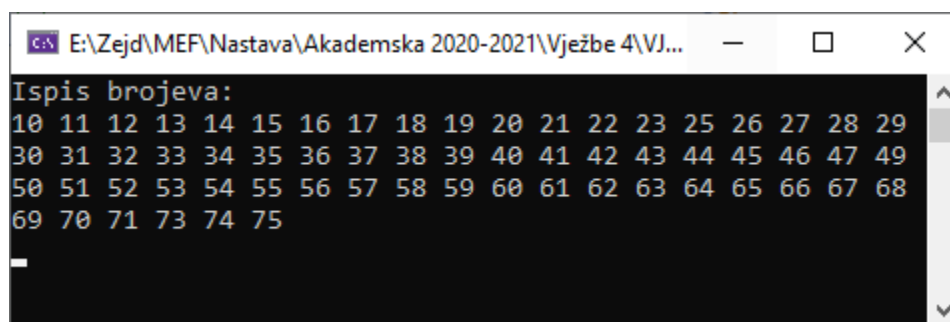
Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vjež...
Ispis brojeva:
10 11 12 13 14 15 16 17 18 19 20 21 22 23 _
```

18) Kreirati program u kojem će se ispisati u konzolnom prozoru svi pozitivni cijeli brojevi od 10 do 75. Ukoliko je jedan od brojeva iz ovog intervala djeljiv sa 8 i sa 3, ne ispisivati takav broj u konzolnom prozoru.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ...
Ispis brojeva:
10 11 12 13 14 15 16 17 18 19 20 21 22 23 25 26 27 28 29
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 49
50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68
69 70 71 73 74 75
_
```

19) Kreirati program koji će simulirati igru u kojoj se od korisnika zahtijeva da pokuša pogoditi skriveni cijeli broj koji se nalazi u intervalu od 1 do 10. Korisnik ima maksimalno tri pokušaja da

pokuša pogoditi skriveni broj i ukoliko uspije u konzolnom prozoru će biti ispisana poruka u vidu čestitke, a ukoliko ne uspije pogoditi skriveni broj u tri pokušaja, program će obavjestiti u konzolnom prozoru korisnika o kojem skrivenom broju se radilo. Tokom igre program će obavještavati korisnika o broju preostalih pokušaja.

Napomena:

- Pretpostaviti da će korisnik unositi cijele brojeve tokom igre.
- U prvoj verziji zadatka vi odredite skriveni broj (npr. nek bude broj 7) koji treba „pogoditi“, a u drugoj verziji omogućiti računaru da on sam odredi svoj skriveni broj kojeg treba pogoditi. Obje verzije je moguće uraditi na više načina koristeći while ili do-while strukture

Ispis u konzolnom prozoru nakon izvršavanja programa:

```
C:\Windows\system32\cmd.exe
Probajte pogoditi skriveni broj koji se nalazi u intervalu od 1 do 10.
Broj preostalih pokusaja: 3
Vas broj je: 6
Uneseni broj 6 nije skriveni broj.

Probajte pogoditi skriveni broj koji se nalazi u intervalu od 1 do 10.
Broj preostalih pokusaja: 2
Vas broj je: 4
Uneseni broj 4 nije skriveni broj.

Probajte pogoditi skriveni broj koji se nalazi u intervalu od 1 do 10.
Broj preostalih pokusaja: 1
Vas broj je: 3
Uneseni broj 3 nije skriveni broj.

Igra je gotova. Skriveni broj je broj 5.
```

```
C:\Windows\system32\cmd.exe
Probajte pogoditi skriveni broj koji se nalazi u intervalu od 1 do 10.
Broj preostalih pokusaja: 3
Vas broj je: 10
Uneseni broj 10 nije skriveni broj.

Probajte pogoditi skriveni broj koji se nalazi u intervalu od 1 do 10.
Broj preostalih pokusaja: 2
Vas broj je: 4

Cestitamo! Uneseni broj 4 je skriveni broj.
```

20) Svaki pozitivni cijeli broj se može redukovati na broj 1 koristeći sljedeći Collatz-ov algoritam:

- ako je broj n paran podijeliti ga sa 2 (tj. $\frac{n}{2}$)
- ako je broj n neparan pomnožiti ga sa 3 i dodati 1 (tj. $3n + 1$)

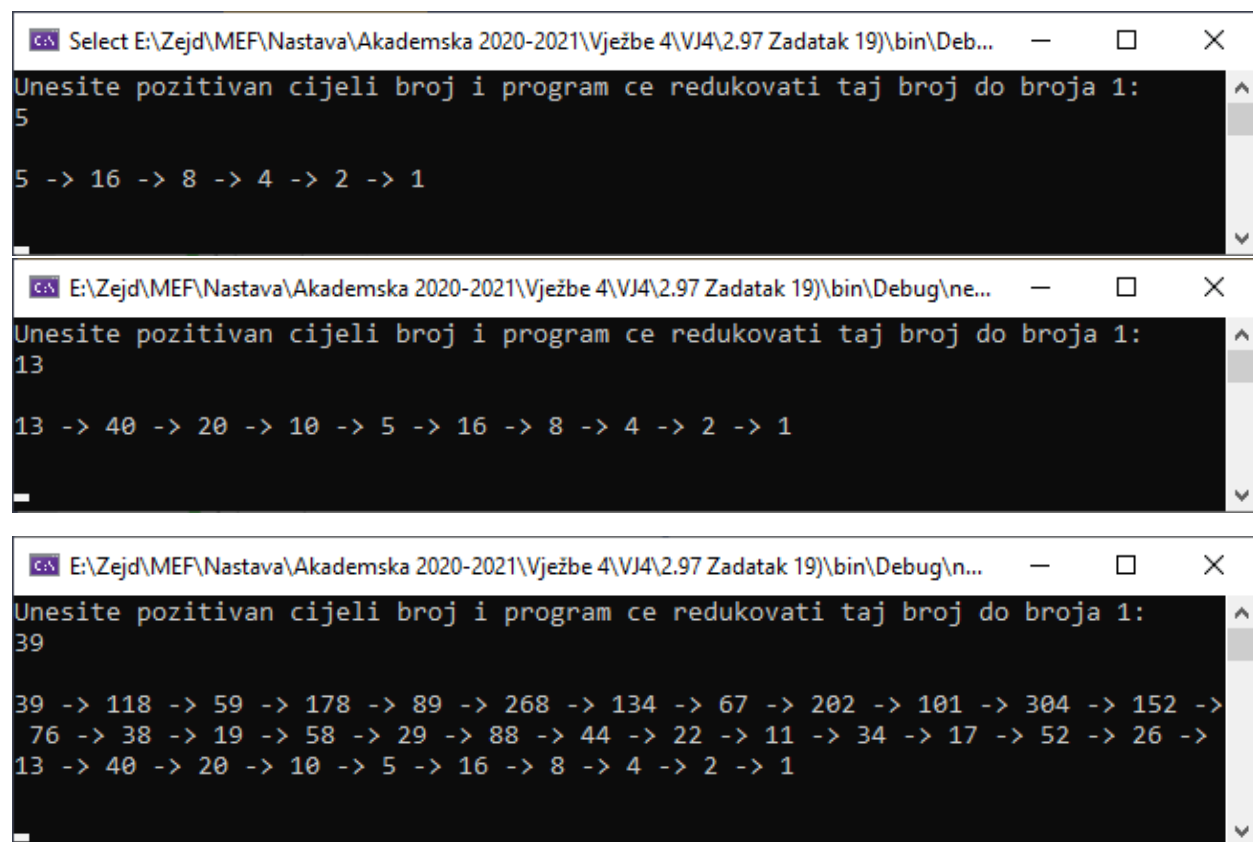
Prethodno definisan algoritam je potrebno ponavljati sve dok broj n ne postane 1.

Kreirati program u C# u kojem će se zahtijevati da korisnik unese jedan pozitivan cijeli broj i program će izvršiti redukciju takvog broja sve do broja 1 koristeći Collatz-ov algoritam. U svakom koraku će se ispisati trenutna vrijednost broja kao u primjerima.

Napomena:

- Pretpostaviti da će korisnik unijeti pozitivan cijeli broj

Ispis u konzolnom prozoru nakon izvršavanja programa:



```

C:\> Select E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.97 Zadatak 19)\bin\Deb...
Unesite pozitivan cijeli broj i program ce redukovati taj broj do broja 1:
5
5 -> 16 -> 8 -> 4 -> 2 -> 1

C:\> E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.97 Zadatak 19)\bin\Debug\ne...
Unesite pozitivan cijeli broj i program ce redukovati taj broj do broja 1:
13
13 -> 40 -> 20 -> 10 -> 5 -> 16 -> 8 -> 4 -> 2 -> 1

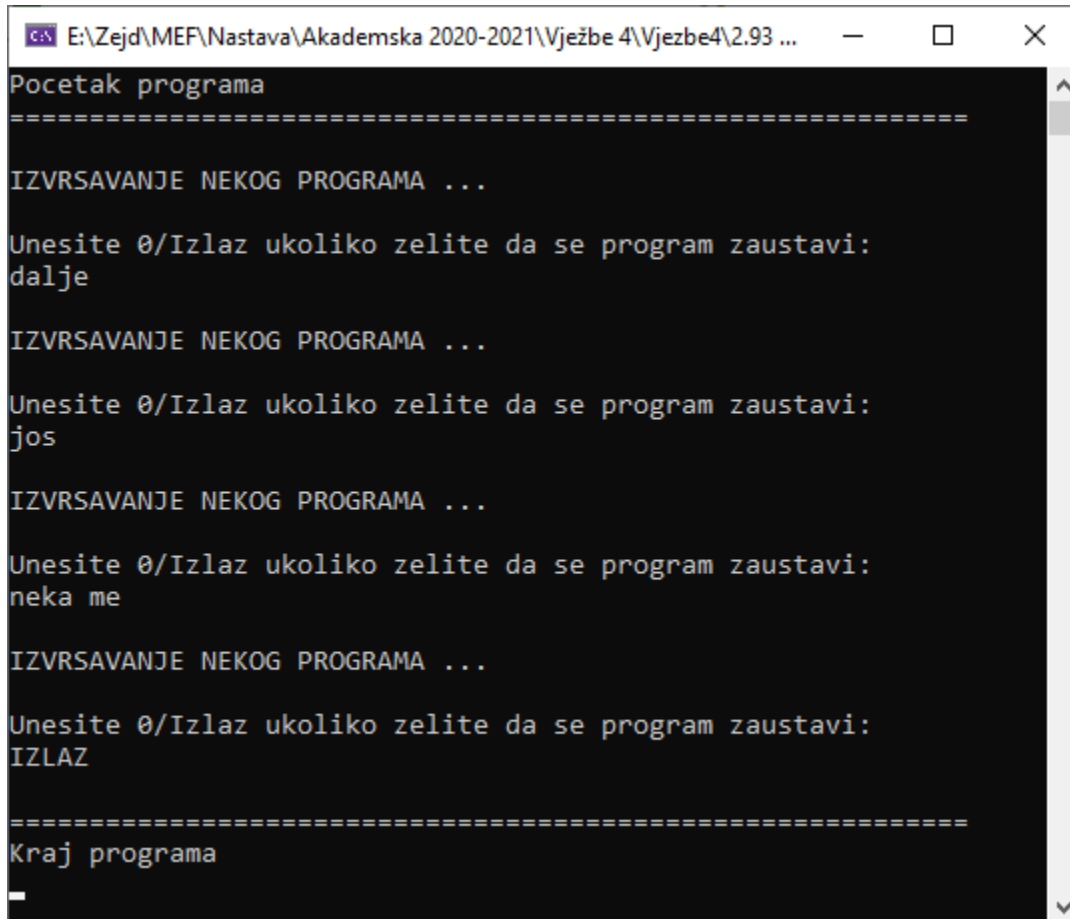
C:\> E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VJ4\2.97 Zadatak 19)\bin\Debug\n...
Unesite pozitivan cijeli broj i program ce redukovati taj broj do broja 1:
39
39 -> 118 -> 59 -> 178 -> 89 -> 268 -> 134 -> 67 -> 202 -> 101 -> 304 -> 152 ->
76 -> 38 -> 19 -> 58 -> 29 -> 88 -> 44 -> 22 -> 11 -> 34 -> 17 -> 52 -> 26 ->
13 -> 40 -> 20 -> 10 -> 5 -> 16 -> 8 -> 4 -> 2 -> 1

```

21) Nekada je potrebno da se program izvršava sve dok korisnik ne unese određenu vrijednost koja zaustavlja rad programa. Definirati program koji će pitati korisnika da li želi da se program

zaustavi. Ukoliko korisnik sa tastature unese broj 0 ili Izlaz (u bilo kojoj kombinaciji velikih i malih slova) program će biti zaustavljen, u suprotnom program će se izvršavati sve dok korisnik ne unese navedene vrijednosti sa tastature. U konzolnom prozoru je potrebno ispisati „početak programa“ na početku programa, „izvršavanje programa...” dok god se program izvršava (program će se izvršiti barem jedanput) i „kraj programa“ kada korisnik unese navedene vrijednosti sa tastature određene za zaustavljanje programa.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\Vježbe4\2.93 ...
Pocetak programa
=====
IZVRSAVANJE NEKOG PROGRAMA ...
Unesite 0/Izlaz ukoliko zelite da se program zaustavi:
dalje
IZVRSAVANJE NEKOG PROGRAMA ...
Unesite 0/Izlaz ukoliko zelite da se program zaustavi:
jos
IZVRSAVANJE NEKOG PROGRAMA ...
Unesite 0/Izlaz ukoliko zelite da se program zaustavi:
neka me
IZVRSAVANJE NEKOG PROGRAMA ...
Unesite 0/Izlaz ukoliko zelite da se program zaustavi:
IZLAZ
=====
Kraj programa
```

Jedan zadatak se može postepeno nadograđivati u više nivoa implementacije kao npr.:

1 - sabiranje svih cijelih brojeva od 1 do 300.

2 - sabiranje svih cijelih brojeva od 1 do broja kojeg definiše korisnik (pretpostavka da će unijeti pozitivan cijeli broj).

3 - sabiranje svih cijelih brojeva od 1 do broja kojeg definiše korisnik (provjeravanje unosa - Defensive Programming koristeći TryParse metode i if-else strukture).

4 - sabiranje svih cijelih brojeva od 1 do broja kojeg definiše korisnik, ostanak u programu dok korisnik ne unese cijeli broj (while struktura koja se izvršava sve dok korisnik ne unese pozitivan cijeli broj)

5 - sabiranje svih cijelih brojeva od 1 do broja kojeg definiše korisnik, ostanak u programu dok korisnik ne unese cijeli broj. Postavlja se upit da li korisnik želi ostati u programu (pretpostavka da će korisnik unijeti samo „DA“ ili „NE“ u svim kombinacijama).

6 - sabiranje svih cijelih brojeva od 1 do broja kojeg definiše korisnik, ostanak u programu dok korisnik ne unese cijeli broj. Postavlja se upit da li korisnik želi ostati u programu i ostanak u ovom dijelu programa dok korisnik ne unese samo „DA“ ili „NE“. (potrebno koristiti dvije while strukture, jednu za unos broja na početku, drugi za unos ostanka ili izlaska iz programa).

Sljedeći zadaci (22- 27) će pratiti navedeni put implementacije dok vi možete bilo koji od prethodnih zadataka na isti način poboljšati samu implementaciju

22) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` koja je jednaka 300. Program nakon sabiranja treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja `<numLimit>` je jednaka `<totalSum>`."

23) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` čiju vrijednost korisnik unosi sa tastature. Program nakon sabiranja treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja `<numLimit>` je jednaka `<totalSum>`."

Napomena:

- Pretpostaviti da će korisnik unijeti pozitivan cijeli broj kao vrijednost varijable `numLimit`.

- Preporuka je da varijabla `totalSum` bude u/long tipa podataka.

24) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` čiju vrijednost korisnik unosi sa tastature i čiju vrijednost treba provjeriti. Ukoliko je korisnik unio bilo šta drugo osim pozitivnog cijelog broja program ispisuje u konzolnom prozoru upozorenje, u suprotnom program nakon sabiranja treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja `<numLimit>` je jednaka `<totalSum>`."

25) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` čiju vrijednost korisnik unosi sa tastature i čiju vrijednost treba provjeriti. Ukoliko je korisnik unio bilo šta drugo osim pozitivnog cijelog broja program ispisuje u konzolnom prozoru upozorenje i ponovno će od korisnika zahtijevati da upiše pozitivan cijeli broj. Nakon što unese pozitivan cijeli broj program treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja `<numLimit>` je jednaka `<totalSum>`."

26) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` čiju vrijednost korisnik unosi sa tastature i čiju vrijednost treba provjeriti. Ukoliko je korisnik unio bilo šta drugo osim pozitivnog cijelog broja program ispisuje u konzolnom prozoru upozorenje i ponovno će od korisnika zahtijevati da upiše pozitivan cijeli broj. Nakon što unese pozitivan cijeli broj program treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja `<numLimit>` je jednaka `<totalSum>`."

Program nakon ispisa postavlja korisniku upit da li želi da ostane u programu i ako korisnik unese „da“ program će ponovno izvršiti akciju, ukoliko unese „ne“ program se prekida.

Napomena:

- Pretpostaviti da će korisnik unijeti ili „DA“ ili „NE“ samo.

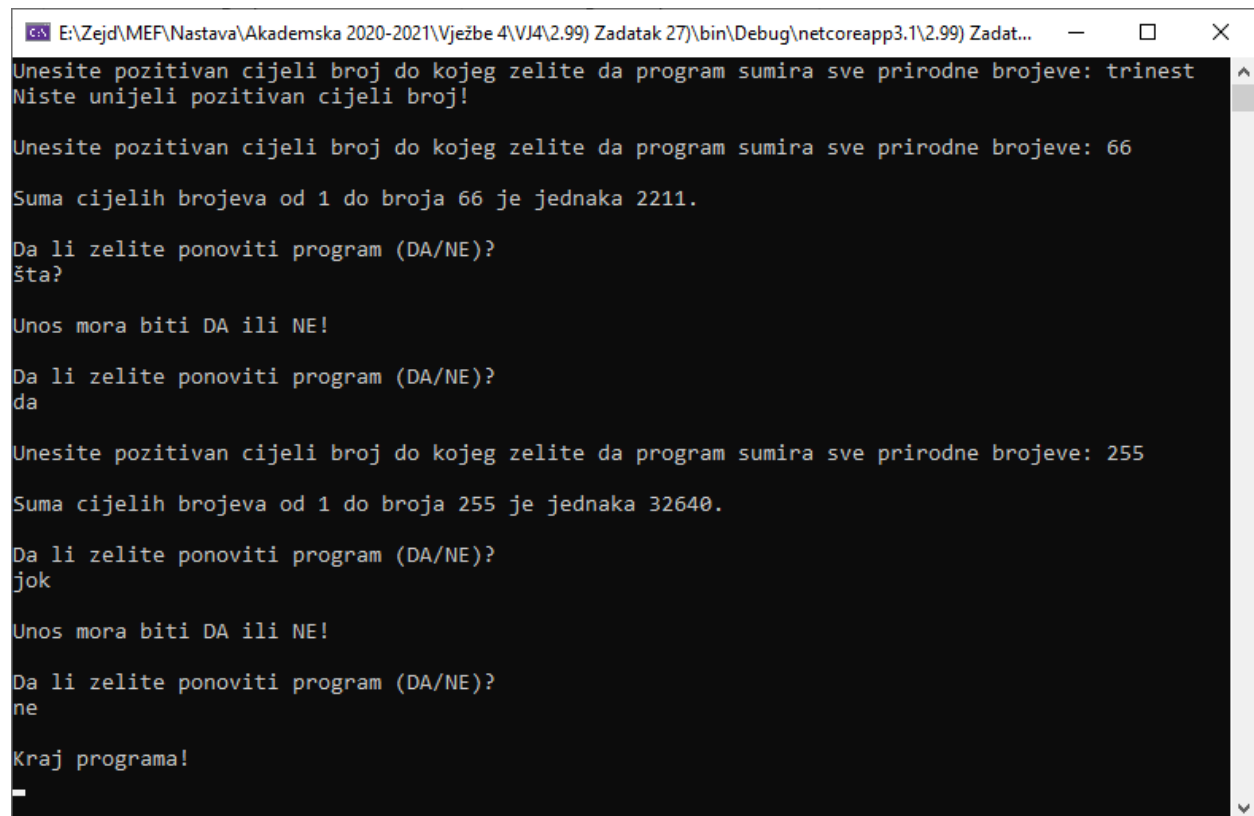
27) Potrebno je kreirati program koji će sabirati pozitivne cijele brojeve od 1 do granične vrijednosti `numLimit` čiju vrijednost korisnik unosi sa tastature i čiju vrijednost treba provjeriti.

Ukoliko je korisnik unio bilo šta drugo osim pozitivnog cijelog broja program ispisuje u konzolnom prozoru upozorenje i ponovno će od korisnika zahtijevati da upiše pozitivan cijeli broj. Nakon što unese pozitivan cijeli broj program treba u konzolnom prozoru da ispiše:

"Suma cijelih brojeva od 1 do broja <numLimit> je jednaka <totalSum>."

Program nakon ispisa postavlja korisniku upit da li želi da ostane u programu i ako korisnik unese „da“ program će ponovno izvršiti akciju, ukoliko unese „ne“ program se prekida. Ukoliko unese bilo šta drugo program izbacuje upozorenje u konzolnom prozoru da unos nije poznat i ponovno postavlja upit sve dok korisnik ne unese „DA“ ili „NE“ u bilo kojoj kombinaciji malih i velikih slova.

Ispis u konzolnom prozoru nakon izvršavanja programa:



```
E:\Zejd\MEF\Nastava\Akademska 2020-2021\Vježbe 4\VI4\2.99) Zadatak 27)\bin\Debug\netcoreapp3.1\2.99) Zadat...
Unesite pozitivan cijeli broj do kojeg zelite da program sumira sve prirodne brojeve: trinest
Niste unijeli pozitivan cijeli broj!

Unesite pozitivan cijeli broj do kojeg zelite da program sumira sve prirodne brojeve: 66
Suma cijelih brojeva od 1 do broja 66 je jednaka 2211.

Da li zelite ponoviti program (DA/NE)?
šta?

Unos mora biti DA ili NE!

Da li zelite ponoviti program (DA/NE)?
da

Unesite pozitivan cijeli broj do kojeg zelite da program sumira sve prirodne brojeve: 255
Suma cijelih brojeva od 1 do broja 255 je jednaka 32640.

Da li zelite ponoviti program (DA/NE)?
jok

Unos mora biti DA ili NE!

Da li zelite ponoviti program (DA/NE)?
ne

Kraj programa!
```