**Programiranje I :: Priprema za parcijalni ispit**

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li je broj prost broj. | |  |  |  | Provjerava da li su cifre broja sortirane rastućim redoslijedom. |   Uraditi implementaciju datih funkcija, a zatim napisati program gdje je potrebno unijeti varijablu:   * cjelobrojna vrijednost za koju važi ograničenje   Ispisati sve proste brojeve u opsegu čije su cifre sortirane rastućim redoslijedom. |
| Rješenje zadatka kopirati u novi fajl (na desktopu) pod nazivom ***Zadatak 1.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li broj sadrži makar jednu parnu cifru. | |  |  |  | Računa sumu cifara broja . |   Uraditi implementaciju datih funkcija, a zatim napisati program gdje je potrebno unijeti varijable:   * cjelobrojna vrijednost za koju važi ograničenje * cjelobrojna vrijednost za koju važi ograničenje   Ispisati sve brojeve u rasponu vrijednosti za koje važi:   * Suma cifara je veća ili jednaka od vrijednosti * Ne sadrže niti jednu parnu cifru |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 2.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Vraća ukupan broj cifara broja . | |  |  |  | Provjerava da li je vrijednost binarni broj (sačinjen samo od nula i jedinica). | |  |  |  | Primjenjuje logičku operaciju ekvivalencije između pojedinačnih cifara ( na odgovarajućim pozicijama) brojeva i po sljedećoj formuli: |   Simbol .  Simbol .  Simbol .  Potrebno je da se pomenute logičke operacije primjene između pojedinačnih cifara koje moraju imati vrijednosti 0 ili 1.  Napisati programski kod za unošenje cjelobrojnih vrijednosti:   * cjelobrojna vrijednost za koju važi da je binarni broj * cjelobrojna vrijednost za koju važi da je binarni broj   **Napomena**: Obavezno provjeriti prilikom unosa da li su i validne binarne vrijednosti, te da li imaju podjednak broj cifara (u suprotnom, ponavljati unos).  Zatim ispisati rezultat funkcije nad unesenim brojevima i |
| **Obzirom da je prilikom prvobitnog unosa vrijednosti unesena nevalidna vrijednost (1012), ponavlja se unos za oba binarna broja.**  Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 3.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li je vrijednost binarni broj (sačinjen samo od nula i jedinica). | |  |  |  | Pretvara binarni broj u decimalni broj. |   Napisati programski kod za unošenje cjelobrojnih vrijednosti:   * cjelobrojna vrijednost za koju važi (   A zatim je potrebno uraditi unos cjelobrojne vrijednosti (**koja mora biti binarni broj**) ukupno puta. Ispisati najveću binarnu vrijednost od svih unosa i njegovu decimalnu reprezentaciju.  **Napomena**: Prilikom unošenja binarnog broja , ponavljati unos dokle god unesena vrijednost nije validna. |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 4.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li je godina prijestupna. Prijestupna godina je ona koja je djeljiva sa 400 ili ona koja je djeljiva sa 4 i nije djeljiva sa 100. | |  |  |  | Vraća broj dana za mjesec u godini . | |  |  |  | Provjerava da li je datum ( validan. | |  |  |  | Provjerava da li je datum 1 ( stariji od datuma 2 (. |   Implementirati date funkcije, a zatim napisati program za unos 3 datuma pri čemu je potrebno voditi računa da se unos pojedinog datuma ponavlja dokle god je taj datum nevalidan. Drugim riječima, potrebno je unijeti 3 validna datuma, nakon čega program treba da ispiše najstariji među njima. |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 5.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li je godina prijestupna. Prijestupna godina je ona koja je djeljiva sa 400 ili ona koja je djeljiva sa 4 i nije djeljiva sa 100. | |  |  |  | Vraća broj dana za mjesec u godini . | |  |  |  | Provjerava da li je datum ( validan. |   Implementirati date funkcije, a zatim napisati program za unos datuma ( pri čemu je potrebno voditi računa da se unos ponavlja dokle god je taj datum nevalidan. Za uneseni datum ( ispisati koji je po redu dan u datoj godini . |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 6.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Provjerava da li je broj binarni broj. | |  |  |  | Konvertuje decimalni (dekadski) broj u odgovarajuću binarnu vrijednost. | |  |  |  | Izračunava sumu cifara za dati broj. |   Implementirati date funkcije, a zatim napisati program za unos sljedećeg:   * cjelobrojna vrijednost koja označava broj puta koliko će se raditi unos (mora biti isključivo pozitivna vrijednost)   A zatim je potrebno uraditi unos cjelobrojne vrijednosti ukupno puta. Ispisati najmanju vrijednost od svih unosa, njegovu binarnu reprezentaciju i sumu cifara njegove binarne reprezentacije. |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 7.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Vraća najveći broj brojnog sistema za broj cifara | |  |  |  | Konvertuje broj iz bilo kojeg brojnog sistema (2,3,4,5,6,7,8,9) u decimalni (dekadski) sistem. Ignorisati heksadecimalni sistem. |   Implementirati date funkcije, a zatim napisati program za unos sljedećeg:   * cjelobrojna vrijednost koja označava broj cifara (mora biti pozitivan)   A zatim je proći kroz sve vrijednosti u opsegu ispisati bazu brojnog sistema, te najveći broj datog brojnog sistema gdje je broj cifara kao i njegov dekadski zapis (koristiti funkciju ). |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 8.cpp*** |

|  |
| --- |
|  |
| Neka su date sljedeće funkcije:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Omogućuje uplaćivanje vrijednosti na bankovni račun i uvećavanje vrijednosti *balance.* | |  |  |  | Omogućuje isplaćivanje vrijednosti sa bankovnog računa i umanjenje vrijednosti . | |  |  |  | Ispisuje trenutno stanje na bankovnom računu. |   Implementirati date funkcije, a zatim napisati program i uraditi sljedeće:   * Inicijalizirati stanje na računu na vrijednost od 0 KM * Omogućiti korisnički unos varijable koja označava odabir funkcionalnosti (1 – Uplata, 2 – Isplata, 3 – Provjera stanja, 4 – Izlaz iz programa)   U slučaju da je unesena opcija koja nije validna, ispisati poruku i ponavljati korisnički unos. Potrebno je napomenuti da se program izvršava sve dok korisnik ne odabere opciju 4. |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 9.cpp*** |

|  |
| --- |
|  |
| Neka je data sljedeća funkcija:   |  |  |  |  | | --- | --- | --- | --- | | **Funkcija** | **Ulazi** | **Izlaz** | **Definicija** | |  |  |  | Vraća broj sačinjen od parnih cifara broja |   Implementirati datu funkciju, a zatim napisati program za unos sljedećeg:   * cjelobrojna vrijednost koja mora biti pozitivna * cjelobrojna vrijednost koja mora biti pozitivna   Prilikom korisničkog unosa voditi računa da je zadovoljen zahtjev A zatim je potrebno proći kroz sve vrijednosti u opsegu i ispisati sve brojeve koji imaju makar jednu parnu cifru **(osim ako je ta jedina cifra 0)**. Ukoliko broj ne sadrži niti jednu parnu cifru, tada se isti ne ispisuje. |
| Rješenje zadatka kopirati u novi fajl pod nazivom ***Zadatak 10.cpp*** |

|  |
| --- |
|  |
| 1. Rješenja zadataka imenovati na sljedeći način:      1. Dodati sve **\*.cpp** fajlove u jedan folder sa nazivom **IB230\*\*\*** (broj indeksa studenta) 2. Uraditi kompresiju    1. Desni klik na folder i odabrati opciju ***Add to archive***    2. Odabrati **zip**kao format kompresije     Rezultat kompresije je fajl sa nazivom *IB210\*\*\*.zip*   1. Priključiti se na ftp server ([ftp.fit.ba](ftp://ftp.fit.ba)) preko File Explorer-a sa sljedećim login podacima:  * Username: **student\_eb** * Password: **student\_eb**  1. Kopirati fajl **IB230\*\*\*.zip** u odgovarajući folder (**Upload** sekcija) |

**Napomena: Da bi se određeni zadatak bodovao, potrebno je da se isti može uspješno pokrenuti i izvršavati. Ukoliko postoji greška prilikom kompajliranja, zadatak se označava kao 'NC' (Not Compiled), te se kao takav ne može bodovati. Isto tako, kardinalnom greškom se smatra nekontrolisano izvršavanje programa poput beskonačne petlje. U tom slučaju se zadatak, označava kao 'RE' (Runtime Error).**