

Tehtävä 2.

Math.sqrt(), n.isnumeric(), import math, float, if/else = standardikirjasto, pythonin sisäisiä funktioita jotka määritelty standardikirjastossa. Import math tuo kirjaston, math.sqrt() käyttää math kirjaston funktiota, ja n.isnumeric testaa syötteen merkkijonon, onko kyseessä numero (int, float ...) vai jokin muu.

Input/Print ja memory = järjestelmätason operaatio, tulostetaan tietoa näytölle tai pyydetään tietoa näppäimistöltä, järjestelmä osallisena pyynnöissä.

Coding = UTF-8 käyttöjärjestelmätason operaatio, nimeää lähdekooditiedoston koodauksen, hakee käyttöjärjestelmästä UTF-8 merkistön?

Tehtävä 3.

Jaetut kirjastot ovat tapa jakaa ohjelmakoodia useiden ohjelmien kesken. Esimerkiksi pythonin koodiesimerkissä käytetään jaettua kirjastoa ctypes, joka tarjoaa C-yhteensopivia tietotyypppejä ja mahdollistaa C-kielelle ominaisten toimintojen kutsumisen. Windows käyttää esim. DLL kirjastoja ja Unix järjestelmät puolestaan SO kirjastoja.

Merkkijonon koodaus tulee muuttaa sillä, jos ASCII encodingia ei olisi ohjelma tulostaisi ainoastaan vain ensimmäisen tavun merkkijonosta. C-kielen printf vaatii merkkijonon tavuina, jotta se voisi tulostaa sen, ASCII merkistökoodauksessa kirjaimet muutetaan numeroiksi. .encoding("ASCII") lisää koodinpätkään muuttaa se merkkijonon tavuiksi ja tulostus onnistuu ctypes kirjastolla.

Kirjoitettaessa tulostusfunktio sys.stdout.write() ohjelma tulostaa tekstin suoraan näyttökonsoliin, eikä se kulje jaetun kirjaston kautta, vaan funktio suoritetaan suoraan ohjelmatiedostosta. Tämän lisäksi merkkijonon muotoa ei tarvitse muokata erikseen sillä python tunnistaa sen itse unicode-merkistön mukaiseksi.

Tulostuspalvelut ovat yleisesti jaettu kolmeen eri tasoon; käyttäjätaso, käyttöjärjestelmä sekä laitteisto ja ajurit. Käyttäjätasolla toimivat esimerkiksi ohjelmointikielet kuten python ja jaetut kirjastot esim pythonin ctypes kirjasto. Nämä ovat luotu helpottamaan ihmisten työskentelyä tuomalla käyttäjälle erilaisia työkaluja

ja välittää käyttäjän kirjoittamia tehtäviä eteenpäin käyttöjärjestelmälle. Käyttöjärjestelmä puolestaan lukee ja hallinnoi käyttäjältä saatuja komentoja ja käskyjä. Ilman käyttöjärjestelmän hallinnointia jokainen ohjelmistokieli pitäisi kääntää suoraan ajurien ja laitteiston luettavaksi, joka käytännössä olisi paljon turhaa työtä. Käyttöjärjestelmä siis itsessään mahdollistaa erilaisten työkalujen käyttämisen eri laitteilla. Lopulta kun käyttöjärjestelmä välittää komennon viimeiselle tasolle, eli laitteistolle ja ajureille ne vievät komennon loppuun. Esim., tehtävänannon esimerkkiohjelma jossa, ensin käyttäjä kirjoittaa koodin, jonka käyttöjärjestelmä lukee ja ohjaa eteenpäin näytön ajureille, jotka valaisevat näytöstä pikseleitä ohjeistuksen mukaisesti. Koostetusti nämä eri tasot mahdollistavat siis laitteille monipuoliset työkalut ilman että ne noudattavat mitään tiettyjä spesifejä vaatimuksia.