Kodavimo standartas

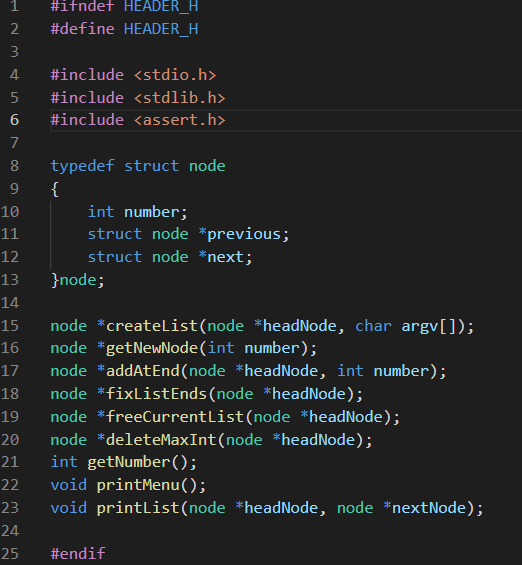
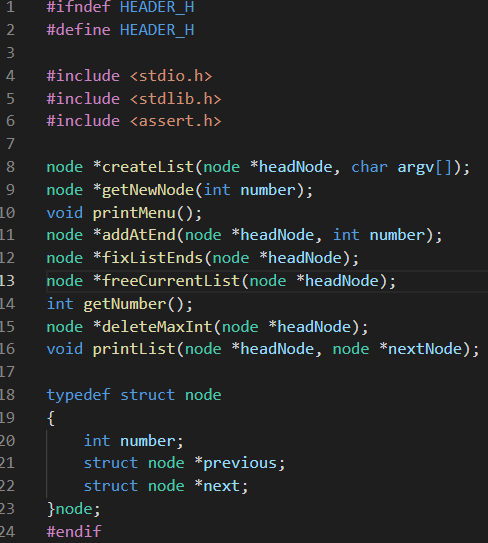
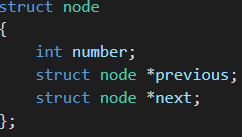
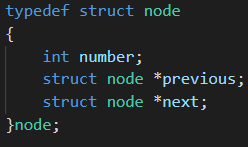
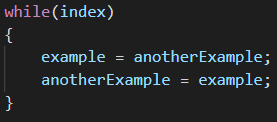
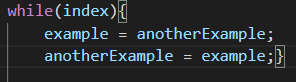
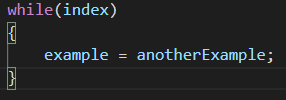
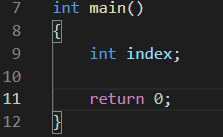
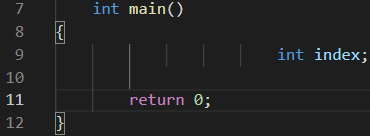
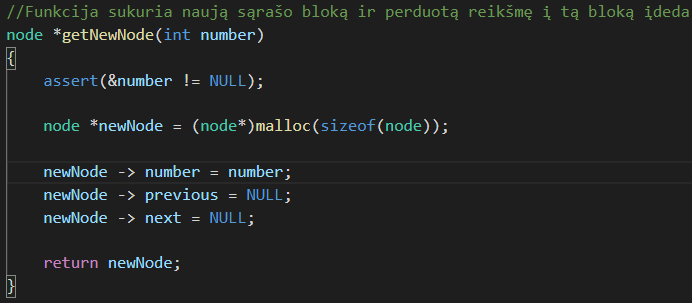
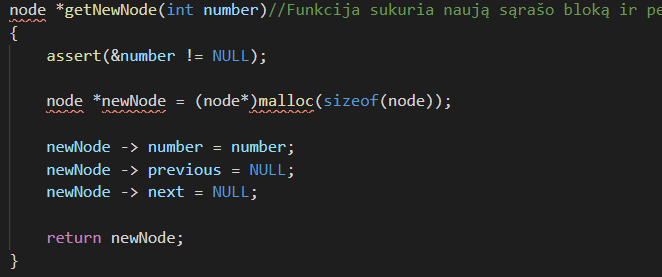
Autorius: Armantas Pikšrys PS3 (LSP: 2016018)

Versija: 1.0

Standarto programavimo kalba: C

Jei pateikiamas kodo pavyzdys, tai kairėje yra tinkamas pavyzdys, o dešinėje blogas.

Aspektai ir jų taisyklės:

1. Programos struktūra:
   1. Failai turi būti skirstomi į atskirus failus, negali būti tik vienas failas, kuriame sudėtos visos funkcijos, aprašai ir t.t. Turi būti .c failas, kuriame yra tik main() funkcija, tada .c failas, kuriame saugomos tik funkcijos ir .h failas kuriame apibrėžiamos funkcijų antraštės ir programoje naudojamos bibliotekos.
   2. Aprašų .h faile pirmiausia aprašome bibliotekas, kurias naudosime, tada struktūras (jeigu jų bus) ir galiausiai funkcijų, kurios iškvieciamos main() funkcijos aprašus, jų tvarka nėra svarbi, bet turėtų būti logiškumas, jei, pavyzdžiui, daug funkcijų darbui su struktūromis yra aprašoma, tai jos turėtų būti šalia viena kitos.
2. Vardai:
   1. Kintamųjų vardai rašomi angliškai, rašomi žodžiai turi reikšmę susijusią su tuo kur tas kintamasis naudojamas programoje, taip pat rašomi pilni žodžiai, išskyrus atvejus, kai norint parašyti pilną žodį be reikalo bus rašoma daug raidžių ir tą žodį sutrumpinus, jo vardas neprarastų reikšmės (PVZ: temp vietoj temporary). Jei varde yra daugiau nei vienas žodis tai pirma raidė mažoji, o kitų žodžių pirmosios raidės didžiosios (PVZ: firstNumber tinka, o FIRSTNUMBER ar firstnumber, ar FirstNumber netinka).
   2. Rodyklės, jei rašoma rodyklė, tai žvaigždutės simbolis rašomas prie pirmosios vardo raidės. Arba adreso simbolis „&“ irgi rašomas prie pat vardo.
   3. Konstantos turi būti aprašomos aprašo faile šitaip PVZ: #define maxChar 255
   4. Struktūrų vardai turi būti aprašomi su typedef, kad kai bus programoje naudojama struktūra, nereikėtų rašyti papildomai „struct“ kiekvieną kartą.
   5. Funkcijų vardai rašomi taip pat kaip ir kintamųjų (exampleFunction). Parametrai, kurie perduodami į funkciją, jos pavadinime (aprašų faile ir funkcijų faile) užrašomi šitaip PVZ: exampleFunction(int number, int \*pointer);
3. Sintaksė:
   1. Skirtukai (while, for, if, else ir t.t.) yra rašomi šitaip, jeigu aprašomas daugiau negu vienas sakinys, tai dedame skirtukus (pirmas skirtukas po funkcija, o antras po paskutiniu sakiniu tos funkcijos). Jeigu tik vienas sakinys, tai skirtukų nededam.
   2. Operatoriai atskiriami tarpais iš abiejų pusių. PVZ: numberOne + numberTwo tinka, bet numberOne+numberTwo netinka.
   3. Kabliataškiai rašomi iš karto po sakinio pabaigos t.y. eilutės pabaigos.
4. Kodo išdėstymas:
   1.  Kodo eilučių įtraukos funkcijose turi būti 1 tab simbolio ilgio nuo esamo įtraukos ilgio. Eilutės, kurios nėra funkcijose neturi įtraukos.
   2. assert() sakiniai funkcijose turi būti atskiriami 1 tarpu iš viršaus ir iš apačios, nebent šalia yra dar vienas assert() sakinys (iš viršaus arba apačios) arba assert() sakinys yra pačioje funkcijos pradžioje.
   3. return ir break sakiniai turi būti atskiriami 1 tarpu iš viršaus (išskyrus atvejus, kai funkcijos bloke return arba break yra vienintelis sakinys).
   4. Funkcijų blokai arba switch sakinio case ir default blokai turi būti atskiriami 1 tarpu iš viršaus ir iš apačios (Išskyrus atvejus, kai jie yra funkcijos pačioje pradžioje arba pačiame gale, arba virš jų yra komentaras).
   5. Visi kiti sakiniai funkcijose gali būti išskiriami pagal autoriaus nuožiūrą (atsižvelgiant į logiką, jei logiška, kad sakiniai atlieka kažkokią procedūrą, kuri yra susijusi su šalia esamais sakiniais, tai jie nėra skiriami tarpais), bet ne daugiau nei 1 tarpu iš viršaus ar apačios.
5. Komentarai:
   1. Komentarai rašomi lietuvių kalba.
   2. Komentarai rašomi prieš funkcijas funkcijų faile ir gali paaiškinti ką ta funkcija atlieka. Tokie komentarai iš viršaus turi turėti 1 tarpą, o iš apačios tarpo neturi, taip pat jų įtrauka tokia pati kaip funkcijos pavadinimo.
   3. Komentarai funkcijų blokuose gali būti rašomi tik tokiu atveju, jei tai leidžia labiau suprasti kokį nors sūdėtingą veiksmą funkcijoje. Jei tai paprastas veiksmas, komentarų nerašome. Komentarai funkcijų blokuose rašomi sakiniui iš viršaus.