

10. Написати функцију која прима један природан број као параметар и на стандардни излаз исписује све природне бројеве мање од тог броја чији запис је палиндром. Функција уједно прима и параметар који говори у ком бројном систему број треба да буде палиндром, али се може само бирати између три бројна система: окталног, декадног и хексадецималног. (Палиндром је низ који је централно симетричан, тј. чита се исто и са лева и са десна)

Код треба да буде писан по C17 стандарду. Дакле, користите све могућности које тај стандард пружа, али и будите у оквиру свих ограничења.

Пројекат мора имати `main.c` датотеку у којој ће бити `main` функција. У `main.c` датотеци одмах на почетку треба да буде блок коментара у којем ћете (у слободној форми) навести опис задатка, кратак опис решења, своје име и презиме и број индекса, као и све важне напомене везане за програм и његово превођење.

Код писати у складу са стандардом кодирања који се налази у датотеци `koding_standard.txt`.

Код треба да буде написан што преносивије. Дакле, да успешно ради и на неким другим платформама, са неким другим компајлерима. Уколико негде морате да уведете неки непреносиви елемент кода важно је да га јасно споменете и објасните на месту примене, као и у коментару на почетку `main.c` датотеке.

У неким задацима се експлицитно спомињу модули (посебне јединице превођења, тј. посебне датотеке са изворним кодом), али трудите се да сами уведете модуле тамо где има смисла иако експлицитно није наведено.

Задаци су намерно формулисани лабаво. Не наводе се конкретни типови података на многим местима (рецимо, кад год спомињемо листу не наводи се ког типа су њени елементи), а и могући број елемената и величине исто нису специфициране. Колико је могуће, учините да програм и његови модули раде што општије, а ви их онда употребите за конкретне примене током тестирања. Ако неке претпоставке морате направити, слободно их направите, али их образложите у коментару на почетку `main.c` датотеке. У вези са тим, размислите добро и о ограничењима вашег програма (нпр. максимална величина и количина података на улазу или структура података и сл.), и напишите их у коментару у `main.c` датотеци. Такође, користите асерте где год сматрате за потребно.

Сви програми треба да буду писани тако да заузимају што мање меморије и да раде што брже, јер увек треба да имате на уму да радимо са системима који имају ограничене ресурсе. У случајевима када су супротстављене тежње да програм буде што мањи и што бржи, одаберите један од та два критеријума и то наведите у коментару у `main.c` датотеци.

Анализирајте да ли је ваш програм написан по MISRA C стандарду. Сасвим је у реду ако нека правила нисте испоштовали у коду, али обавезно то прокоментаришите и јасно назначите у коментару на почетку `main.c`.

Посебну пажњу посветите тестирању. То исто носи део оцене. Обично није довољно само пробати за неколико улаза и сматрати да је тестирање готово. Поступак тестирања који сте обавили морате образложити.