# Baigiamasis testas

**WWW \_\_\_\_\_\_Gabrielius Mažeikis\_\_\_\_\_**

1. ***Ką reiškia WWW? (1)***

\_\_\_\_\_\_\_\_\_\_World Wide Web – informacijos sistema kur interneto resursai yra identifikuoti ir pasiekiami naudojant URL(Uniform Resource Locators – web adresais) ir yra pasiekiami visame internete.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ***Keliais sakinias parašykite, kas yra internetas: (2)***

# Internetas – Pasaulinis komp. įrenginių tinklas jungiantis daug mažesnių įrenginių tinklų, leidžiantis komunikuoti milijonams įrenginių visame pasaulyje.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ***Kas yra HTTP protokolas, kam naudojamas ir kokius tipus žinote? Pailiustruokite piešiniu.(3)***

|  |
| --- |
|  |
| HyperText Transfer Protocol (HTTP) – tai protokolas/metodas skirtas pasiekti svetaines, resursus ir informaciją internete(WWW). HTTP skirtas standartizuoti puslapių duomenų pateikimo ir skaitymo procesą. Su HTTP susiję protokolai(-ų tipai): HTTPS (saugusis HTTP), IMAP, FTP, MIME, SSH ir t.t., visi naudojami informacijos perdavimui per kompiuterių tinklą. |
|  |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ***Kam skirtas GET, o kam POST HTTP requestai? (2)***

GET HTTP metodas užklausia nurodyto URL adreso tam tikro resurso. POST metodas užklausia nurodyto URL adreso apdoroti užklausoje suteiktą (formatą atitinkančią) informaciją, ir kurti naujus resursus.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ***Keliais sakiniais parašykite kas yra Frontend (Client side) ir Backend (server side)? (2)***

Frontend – tai sistemos prezentacinis sluoksnis, vartotojo sąsaja, dažniausiai programuojama naudojant HTML, CSS ir JavaScript.  
  
Backend - sistemos duomenų prieigos sluoksnis kurį sudaro serveris(-iai), kuris(-ie) suteikia reikiamus duomenis užklausos į juos metu, duomenų bazė(-s) kuri(-os) organizuoja informaciją, programa(-os) kuri(-os) duomenis reikiamai procesuoja/perduoda, ir kt. vidinės programos. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# /10

**HTML ir CSS**

6.

**Kokia bus**

**žodžio**

‘

**galimybe’ spalva**

a.

**Žalia**

b.

Mėlyna

c.

Juoda

d.

Niauktai melsvai chaki

7.

**Pagal kokį kriterijų**

**HTML elementuose**

**formuojame <form>**

**’os**

**duomenis**

?

a.

n

ame

b.

p

laceholder

c.

**i**

**nput**

d.

i

d

8.

**K**

**oki**

**ų tipų**

**HTTP užklausas galime išsiųsti**

**formos deka?**

a.

**Post**

b.

Delete

c.

**Get**

d.

Update

9.

**Kokia**

**kita**

**savybė būtina, norint**

**taikyti CSS**

**savybę float?**

a.

**d**

**isplay: block**

b.

position: relative

c.

clear

-

fix: left

10.

**K**

**oks yra numatytasis (default) poz**

**icionavimo**

**tipas?**

a.

left

b.

top

c.

relative

d.

**static**

e.

center

**/**

**10**



1. **HTML tai: (1 atasakymas)** 
   1. Programavimo kalba skirta kurti funkcionalumui internete.
   2. Programavimo kalba žymėms kurti.
   3. **Hierarchija susijusių komponentų tekstinio žymėjimo kalba.**
   4. Nu tai ten ta nesamonė kur rašai kodus internete.
2. **Kur aprašomas HTML dokumento pavadinimas:** 
   1. <body>
   2. **<head>**
   3. <document>
   4. <meta>
3. **CSS tai: (2 atsakymai)** 
   1. Čia kur parašai ‘console.logout’ ir reik per F11 atsidarius pažiūrėt.
   2. **Tai stiliaus savybių priskyrimo, pagal žymėjimo pasirinktis formatas.**
   3. Lygiavimo, formatavimo ir stilizavimo savybių priskyrimo formatas HTML elementams.
   4. Programavimo kalba skirta kurti funkcionalumui internete.
4. **Koks atributas naudojamas nuorodoje <a>?** 
   1. link
   2. src
   3. **href**
   4. url

1. **Pasirinkite input elemento tipus:** 
   1. **text**
   2. **password**
   3. **email**
   4. **color**
   5. colour
   6. kolaur
   7. **number**
   8. yourNumber
   9. **radio**

## Javascript

1. **Parašykite bent 3 javascript panaudojimus?(1)** 
   * Tinklalapius kurti **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**
   * Web programas kurti **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**
   * Serverių aplikacijas kurti **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**
2. **Vienu sakiniu išsireiškite, kas yra framework’as?(1)**

## Tai yra kolekcija kodo bibliotekų suteikianti programuotojams iš anksto parašytus metodus skirtus supaprastinti/pagreitinti darbą. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Išvardinkite 3 javascript Framework’us.(1)**

Node.js, Vue.js, AngularJS **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kokius kintamųjų tipus žinote (atminties naudojimo ir reikšmių prasmėmis)?(2)**

Number, String, Boolean, Undefined, Null, Object, Array, Function **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kuom skiriasi objektas nuo masyvo ir kam jie skirti?(3)**

Objektas yra kintamas specialus duomenų tipas kuris leidžia talpinti jame   
(-apibūdina) įvairius duomenų rinkinius, o masyvas yra specialus kintamojo tipas kuris leidžia jame talpinti kintamųjų sąrašą. Abi duomenų talpyklas galima naudoti įvairiomis paskirtimis kuriant aplikacijas. Objektas ir masyvas yra reference type kintamieji. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kam naudojami prototipai, ir kuom jie skiriasi nuo paprastų objektų?(3)**

Tai yra mechanizmas kurio pagalba objektai paveldi savybes iš vieno kito. Prototipas yra objekto šabloninis-tėvinis objektas kurio savybes pradinis (paprastas) objektas paveldi. Prototipas taip pat gali turėti prototipą kuris irgi, pvz., turi savo prototipus ir t.t.; Kiekvieno objekto prototipas (1+n eilės) yra Object (turintis įvairius default metodus, pvz. valueOf()). **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kas yra event’as ir eventListener’is?(2)**

Eventas (pvz. paspaudimas) yra įvykis kuris nutinka HTML elementuose. eventListener‘is yra metodas kuris laukia kol ivyks Eventas, ir tada informuoja Event sąsąją implementuojantį objektą. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kokius event’us žinote?(1)**

Onclick, onload, onkeydown, onchange, onmouseover, onmouseout

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Kas yra funkcija? Kas yra parametrai? Aprašas? Kvietinys? Grąžinimas?(3)**

## Funkcija yra programos kodas kuris atlieka tam tikrą užduotį. Funkcija „suveikia“ kada ji būna iškviečiama.

## Funkcijos parametrai tai programos elementai (dažniausiai kintamieji) nurodyti funkcijos pavadinime (šalia vardo).

## Funkcijos aprašas yra funkcijos prototipas – išankstinis aprašas(kas turi vykti funkcijoje) ir/arba realizavimo aprašymas (programos realizacijos kode, kur iškviečiama funkcija).

## Funkcijos kvietinys tai kreipinys į funkciją jos vardu (skliaustuose gali būti pateikiami faktiniai parametrai).

## Grąžinimas tai duomenys/rezultatai kurie yra gražinami į programos realizaciją, kai funkcija baigia savo darbą. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**10. Parašykite kam skirti masyvo iteraciniai metodai, pateikite pavyzdžių: (3)**

1. Filter:

Sukuria naują masyvą (iš kito masyvo) kur pateikti elementai atitinka nurodytą salygą, pvz:

pazymiai = [4, 10, 2, 9];

patenkinami = pazymiai.filter(pazymys => pazymys > 5);

console.log(patenkinami);

//10, 9

1. Map

Sukuria naują masyvą (iš kito masyvo) kur iškviečiama funkcija kiekvienam pateiktam elementui:

pazymiai = [4, 10, 2, 9];

patenkinami = pazymiai.map(pazymys => pazymys + 1);

console.log(patenkinami);

//5, 11, 3, 10

1. Reduce –

Kiekvienam duoto masyvo elementui iškviečia bendrą funkciją ir gale grąžina bendros funkcijos akumuliuotą(visų elementų kuriems buvo iškviesta funkcija), vieną-bendrą funkcijos reikšmę.

pazymiai = [4, 10, 2, 9];

suma = pazymiai.reduce ((bendrasum, pazymys) => bendrasum + pazymys);

vidurkis = (suma / pazymiai.length);

console.log(vidurkis);

//6.25

**/20**

### PHP (Atsakyti atskiram lape)

1. **Kas yra klasė? (2) –** Tai sudėtinis duomenų tipas kuriame yra aprašyti metodai ir kintamieji.
2. **Kas yra objektas? Kaip objektas susijęs su konstruktoriumi? (3) –** Tai sudėtinio duomenų tipo (klasės) instancija. Objekto konstruktorius yra iškviečiamas kada objektas yra sukuriamas proceso metu. Konstruktorius aprašo, kas turi įvykti su objekto parametrais arba pačia programa, kada objektas sukuriamas.
3. **Kuom skiriasi funkcijos perkrovimas (overload) nuo funkcijos perrašymo (override)?(3)**

Funkcijos perkrovimo atveju, daugiau nei viena funkcija gali turėti tą patį pavadinimą, tačiau priimti skirtingą kiekį parametrų (naudojama kai programoje reikalinga funkcija kur yra pateikiami skirtingi kiekiai kintamųjų). Funkcijos perrašymo atveju, daugiau nei viena funkcija turės tą patį pavadinimą ir tą patį parametrų kiekį (naudojama perrašant tėvinės klasės funkciją vaiko klasėje).

1. **Kas yra paveldimumas, kokios pagrindinės jos loginės sąlygos?(4)**

Klasių paveldimumas yra kada viena klasė paveldi-pasisavina kitos klasės parametrus ir metodus. Sąlygos:

* Vaiko klasė yra apibrėžta su extends žodeliu (class Vaikas extends Tevas)
* Vaiko klasė gali pasinaudoti arba perrašyti (override) tėvo klasės parametrus/metodus
* Tėvo klasės parametrai/metodai, kuriuos norima paveldėti yra apibrėžti public arba protected prieigos salygomis
* Tėvo klasės apibrežime nepanaudotas žodelis final, kuris uždraudstų kitų klasių paveldėjimą

1. **Kas yra polimorfizmas? Ir kokiam tikslui jis naudojamas?(4)**

Polimorfizmas yra objektinio programavimo logika, kuriame klasės gali turėti skirtingus funkcionalumus, tačiau dalintis bendra sąveika (arba abstraktine klase). Polimorfizmas naudojamas kai skirtingų klasių objektai turi atlikti skirtingus uždavinius, tačiau jie dalinasi bendrais metodų vardais (pvz. Mechaninė extends PavarųDežė, Automatinė extends PavarųDežė, ir automatinė/mechaninė klasės turi metodą „permestiPavarą“, kuris skirtingose dežėse skirtingai persimeta).

1. **Kas yra inkapsuliacija? Išvardinkite bent 3 priežastis kodėl ji reikalinga dideliuose projektuose.(4)**

Inkapsuliacija tai objektinio programavimo logika, kur vidinės objekto detalės yra paslepiamos nuo išorinių šaltinių. Išoriniams šaltiniams teikiami duomenys yra prieinami prie apsaugotų (private/protected t.t.) duomenų naudojant GET ir SET metodus.

Dideliuose projektuose naudojama kada:

* Norima apsaugoti klasės duomenis nuo kitų (nekontroliuojamų) projekto/programos dalių, kurios gali bandyti prie jų prieiti arba ją paveldėti
* Lengviau pakeisti vidinį klasės kodą nesugadinant kitų programos dalių, t.y. mažesnė išorinė priklausomybė
* Jei norima iš išorinių šaltinių klasės laukus padaryti tik perskaitomus, galima neapibrėžti SET metodo
* Jei norima iš išorinių šaltinių klasės laukus padaryti tik įrašomus, galima neapibrėžti GET metodo
* Galima atlikti tam tikrus papildomus veiksmus GET/SET apraše kai kitos programos dalys daro GET/SET užklausas.

1. **Kas yra abstrakcija? Kokie apribojimai įsigalina vaikinėms, ir kokie pačiai klasei?(4)**

Abstrakcija tai objektinio programavimo logika, kada tėvinė klasė turi bendrai pavadintus metodus, bet reikalauja vaikinių klasių kad juos implementuotų ir savaip įvykdytų jų užduotis (laikantis reikalavimų).

Ribojimai:

* Kada vaikinė klasė paveldi tėvinės abstraktinės klasės metodą, vaikinės klasės metodas turi turėti tą patį pavadinimą
* Tą patį (arba mažiau griežtą) prieigos parametrą (access modifier).
* Metodo parametrų (įvesties) kiekis turi būti taspats arba didesnis (galimi papildomi parametrai vaikinėje klasėje, kai tėviniai parametrai jau apibrėžti, be to, galimi skirtingi parametrų vardai paveldėjusių klasių implementacijoje)

1. **Kas yra MVC koncepcija? Ką reiškia kiekviena raidė ir kodėl pats MVC modelis yra naudingas?(4)**

MVC - tai programų architektūra/kūrimo būdas kuris logiškai atskiria projekto verslo modelį, programos duomenis ir vartotojo sąsają. MVC naudingas kuriant vartotojo sąsają, kurioje verslo ir programos logika/vidiniai duomenys yra atskirti nuo to, ką vartotojas mato.

M – Modelis (model) – dirba su vidiniais duomenimis kuriuos reikia pateikti su duomenų baze susijusių procesų metu, tokių kaip paieška ar grąžinimas.

V – Vaizdas (view), vartotojo sąsaja, t.y. tai ką mato vartotojas ir su kuo jis sąveikauja (prezentacinis sluoksnis, pvz. HTML, CSS)

C – Kontroliuotojas (Controller) – Verslo logika kuri jungia Modelio ir Vaizdo koncepcijas, yra atsakingas už užklausų iš Vaizdo pusės gavimą, kontroliavimą ir perdavimą į Modelio pusę dėl reikiamo atsako, bei atsakų gavimą ir pateikimą atgal Vaizdui.

1. **Kokius žinote SuperGlobal kintamuosius? Kokia jų paskirtis? Pateikite 2 panaudojimo pavyzdžius.(4)**

SuperGlobal kintamieji yra kintamieji kurie yra prieinami visame kode, visada.

$GLOBALS – kintamasis (masyvas), kuris gali saugoti ir iškviesti nurodytą kintamąjį bet kur kode. Pvz:

function duKartDu() {

$GLOBALS['atsakymas'] = 2 \* 2; }

duKartDu();

echo $atsakymas; // $atsakymas prieinamas visur, ne tik duKartDu funkcijoje

$\_SERVER – kintamasis (masyvas) kurie laiko svetainės ir serverio aplinkos informaciją, pvz:

echo $\_SERVER['PHP\_SELF']; //grąžina dabartinio PHP failo pavadinimą ir vietą (skaičiuojant nuo pagrindinio aplanko).

// Gabrielius\_Mazeikis\_FS-Testas.docx/SitasPavyzdys.php :)

$\_GET – kintamasis skirtas surinkti HTML formos informaciją kurioje naudojamas metodas get. GET naudojamas peržiūrėti informaciją jos nekeičiant.

$\_POST – kintamasis (masyvas) naudojamas rinkti informaciją po metodo post. POST naudojamas peržiūrėti informaciją ją keičiant.

1. **Kas yra sausainiai? Kas yra sesija, ir kuomet ji nutraukiama?(3)**

Sausainiai (globalus k. - $\_COOKIE) yra failai kuriuos serveris įrašo į vartotojo įrenginį, kad identifikuotų unikalaus vartotojo informaciją ir duomenis. Vartotojas ir serveris dalinasi sausainiais kaskart kai atliekamos naujos užklausos.

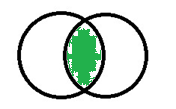
Sesija (globalus k. - $\_SESSION), panašiai kaip sausainiai, saugo informaciją apie vartotoją, tačiau yra užbaigiama (dažniausiai) kada vartotojas palieka svetainę/programą, ir jos informacija yra saugoma serverio pusėje.

# /35

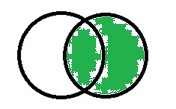
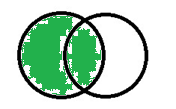
## SQL (Atsakyti atskiram lape)

1. **Nuspalvinkite atitinkamai: (2)**

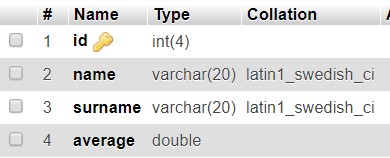
**Full Join Inner join(standard)**



**Left Join Right join**



1. **Parašykite SQL užklausą, kuri grąžintų pirmų dešimties studentų vardus ir pavardes, kurių vidurkis didesnis nei 8. Išrikiuotus didėjimo tvarka (pradedant nuo mažiausio vidurkio) (5)**



SELECT name, surname

FROM studentai(?)

WHERE avarage > 8

ORDER BY avarage ASC

LIMIT 10;

# /7