Šifra kandidata:

Državni izpitni center

M 1 6 2 4 5 1 1 2

JESENSKI IZPITNI ROK

INFORMATIKA

Izpitna pola 2

Sobota, 27. avgust 2016 / 90 minut

Dovoljeno gradivo in pripomočki:
Kandidat prinese nalivno pero ali kemični svinčnik in računalo.
Kandidat dobi dva konceptna lista in ocenjevalni obrazec.

SPLOŠNA MATURA

NAVODILA KANDIDATU
Pazljivo preberite ta navodila.

Ne odpirajte izpitne pole in ne začenjajte reševati nalog, dokler vam nadzorni učitelj tega ne dovoli.

Prilepite kodo oziroma vpišite svojo šifro (v okvirček desno zgoraj na tej strani in na ocenjevalni obrazec). Svojo šifro vpišite tudi na konceptna lista.

Izpitna pola vsebuje 6 nalog. Število točk, ki jih lahko dosežete, je 44. Za posamezno nalogo je število točk navedeno v izpitni poli.

Rešitve, ki jih pišite z nalivnim peresom ali s kemičnim svinčnikom, vpisujte **v izpitno polo** v za to predvideni prostor. Kadar je smiselno, narišite skico, čeprav je naloga ne zahteva, saj vam bo morda pomagala k pravilni rešitvi. Pišite čitljivo. Če se zmotite, napisano prečrtajte in rešitev zapišite na novo. Nečitljivi zapisi in nejasni popravki bodo ocenjeni z 0 točkami. Osnutki rešitev, ki jih lahko napišete na konceptna lista, se pri ocenjevanju ne upoštevajo.

Zaupajte vase in v svoje zmožnosti. Želimo vam veliko uspeha.

Ta pola ima 16 strani, od tega 3 prazne.

M 1 6 2 4 5 1 1 2 0 2

V sivo polje ne pišite.

V sivo polje ne pišite.

1. Peter je opazil, da mu novi računalnik pri delu z več programi hkrati deluje zelo počasi.

1.1 Kaj bi bil lahko vzrok počasnejšega delovanja?

_____

(1)

Razložite, zakaj zapisani vzrok vodi k počasnejšemu delovanju računalnika.

_____

_____

_____

_____

(1)
(2 točki)

1.2 Pri navedenih lastnostih komponent računalnika ugotovite, ali vplivajo na hitrost njegovega delovanja.

Svoje odločitve utemeljite.

Hitrost dostopa do podatkov na disku: _____

_____

(1)

Hitrost mikroprocesorja: _____

_____

(1)

Velikost pomnilnika ROM: _____

_____

(1)
(3 točke)

2. Peter Zmeda redno skrbi za defragmentiranje in čiščenje trdega diska v svojem računalniku. Nekoč mu je program za čiščenje diska javil, da je našel 36.152.320 bitov podatkov v eni ali več datotekah. Petru se niti ne sanja, za kakšne podatke gre, predvideva pa, da gre za tekstovno datoteko, za zvočno datoteko ali za datoteke z barvnimi slikami.

2.1 Izračunajte, koliko znakov vsebuje datoteka, če je to tekstovna ASCII-datoteka.

_____

(1 točka)

2.2 Izračunajte, kako dolg je zvočni posnetek, če privzamemo, da je to nestisnjena zvočna datoteka, vzorčena s 44,1 kHz, 16 biti in dvema kanaloma (stereo).

_____

_____

(1 točka)

2.3 Kaj pa, če je datoteka shranjena v obliki MP3, ki približno desetkratno stisne izvirni zvočni posnetek? Približno kako dolg je zvočni posnetek v tem primeru?

_____

_____

(1 točka)

2.4 Morda pa gre za slikovne datoteke. Izračunajte, največ koliko in najmanj koliko slikovnih datotek je to, če upoštevate, da so slikovne datoteke ustvarjene z digitalnim fotoaparatom v obliki RAW (BMP). Najmanjša velikost fotografij je 2208 × 1473, največja velikost pa 7360 × 4912.

_____

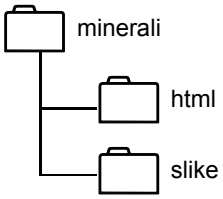
_____

_____

(2 točki)

V sivo polje ne pišite.

3. Peter izdeluje spletne strani o mineralih. Vse datoteke html shranjuje v podmapo ***html***, vse slike pa v podmapo ***slike***. Oblike posameznih elementov, značk, je določil v datoteki *slogi.css*. Ta datoteka je v mapi ***minerali***. Struktura map je predstavljena na spodnji skici.



3.1. Eno od datotek *minerali.html* je zapisal, kot je prikazano spodaj.

Popravite napake v datoteki tako, da bo stran delovala na katerem koli računalniku, ko bo mapo *minerali* kopiral na spletni strežnik.

```
<html>
<head>
   <meta charset="UTF-8">
   <title>Minerali</title>
</head>
<body>
   <link rel="stylesheet" href="slogi.css">
   <h1>Slike mineralov</h1>
   <table cellpadding=5 cellspacing=5 border=2>
      <tr>
         <td><img src="C:\minerali\slike\ahat.jpg"> </td>
         <td> <img src=" C:\minerali\slike\ametist.jpg"></td>
         <td> <img src=" C:\minerali\slike\sljuda.jpg"></td>
         <td> <img src=" C:\minerali\slike\lazurit.jpg"></td>
      <tr>
         <td>Ahat </td>
         <td>Ametist</td>
         <td>Sljuda</td>
         <td>Lazurit</td>
      </tr>
</body
</html>
```

(4 točke)

4. Peter Zmeda ima strica, ki vodi podjetje za izposojo avtomobilov. Za vsak avtomobil ima zbrane te podatke: registrska številka, razred avtomobila, znamka in tip avtomobila ter število sedežev. Ko stranka prvič prevzame avtomobil, da te podatke: ime in priimek, naslov, datum rojstva in številka vozniškega dovoljenja, nakar dobi člansko kartico s temi podatki in člansko številko. Nato prevzame avtomobil. Pri vsakem naslednjem prevzemu avtomobila pokaže samo kartico in že ga lahko prevzame.

   Peter Zmeda je stricu obljubil, da mu bo izdelal podatkovno bazo za enostavno upravljanje podatkov o izposoji avtomobilov. Pri tem pa potrebuje vašo pomoč.

   4.1. Narišite globalni podatkovni model za izposojo avtomobilov.

(2 točki)

   4.2. Narišite osnovni podatkovni model E-R za izposojo avtomobilov in določite številnost posameznega oz. posameznih razmerij. Upoštevajte, da želi stric voditi celotno zgodovino vseh izposoj. Če je številnost razmerja N : N, ga razrešite z uvedbo nove entitete (*normalizacija*).

(2 točki)

4.3. Zapišite vse atribute za vsako od entitet, ki so potrebni, da bo lahko vodil evidenco izposoje. Določite primarne in morebitne tuje ključe.

(6 točk)

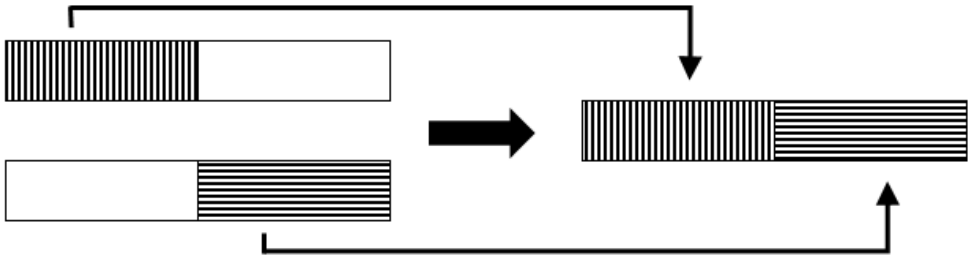
5. Znanstveniki v butalskem državnem laboratoriju iščejo zdravilo proti bakteriji, ki napada butalske koprive. Lastnosti bakterije določa njena DNK, ki sestoji iz štirih različnih nukleotidov: A (adenin), C (citozin), G (gvanin) in T (timin). Poleg tega so nukleotidi definirani v dveh komplementarnih parih:

   - A in T: A je komplement T in obratno,
   - C in G: C je komplement G in obratno.

   Znanstveniki so ugotovili, da DNK bakterije vsebuje 3000 nukleotidov (DNK bakterije je torej niz 3000 znakov, pri čemer ima prvi indeks 0 – če rešujete nalogo v programskem jeziku Pascal, pa naj bo indeks 1). Pri odkrivanju zdravila si znanstveniki pomagajo tako, da poženejo različne računalniške simulacije križanja in mutacij bakterije ter opazujejo, kaj se pri tem dogaja z DNK bakterije.

   Znanstveniki za izvajanje računalniške simulacije potrebujejo te funkcije:

   - `prestej_nukleotide(dnk, nukleotid)` – funkcija vrne število pojavitev nukleotida `nukleotid` v DNK `dnk`.
   - `je_veljavno_zaporedje(dnk)` – funkcija preveri, ali je dano zaporedje `dnk` veljavno zaporedje DNK, torej ali res vsebuje samo nukleotide A, C, G in T.
   - `komplement(nukleotid)` – funkcija vrne komplement nukleotida `nukleotid`, kot je definirano v komplementarnem paru.
   - `mutacija(dnk, indeks, nukleotid)` – funkcija v DNK `dnk` na mestu `indeks` spremeni nukleotid v `nukleotid`.
   - `krizanje(dnk1, dnk2)` – funkcija vrne nov DNK, v katerem je prva polovica zaporedja nukleotidov iz DNK `dnk1` in druga polovica zaporedja nukleotidov iz DNK `dnk2`.



   Odgovorite na spodnja vprašanja. Pri podvprašanjih a) pri vsaki nalogi je DNK dolg 6 znakov. Pri podvprašanjih b) pa upoštevajte, da je DNK dolg 3000 znakov, kakor piše v besedilu naloge.

   5.1. a) Kaj vrne klic funkcije `prestej_nukleotide('ATCGGC', 'G')`?

   _____

   (1)

b) Napišite funkcijo `prestej_nukleotide(dnk, nukleotid)`.

(1)
(2 točki)

5.2. a) Kaj vrne klic funkcije `je_veljavno_zaporedje('ATBGGC')`?

_____

(1)

b) Napišite funkcijo `je_veljavno_zaporedje(dnk)`.

(1)
(2 točki)

5.3. a) Kaj vrne klic funkcije `komplement('A')`?

_____

(1)

b) Napišite funkcijo `komplement(nukleotid)`.

(1)
(2 točki)

5.4. a) Kaj vrne klic funkcije `mutacija('ATCGGC', 2, 'T')`?

_____

(1)

b) Napišite funkcijo `mutacija(dnk, indeks, nukleotid)`.

(1)
(2 točki)

V sivo polje ne pišite.

5.5. a) Kaj vrne klic funkcije `krizanje('ATCGGC', 'ATAGCG')`?

_____

(1)

b) Napišite funkcijo `krizanje(dnk1, dnk2)`.

(1)
(2 točki)

6. Recimo, da imamo podatkovno strukturo tabelo števil: 90, 64, 50 in 78. Če jih želimo urediti od najmanjšega do največjega, jih moramo zapisati kot tabelo števil: 50, 64, 78 in 90. Pri tem opazimo, da se je število 90 premaknilo na mesto z indeksom 3, število 64 na mesto z indeksom 1, število 50 na mesto z indeksom 0 in število 78 na mesto z indeksom 2 (v programskem jeziku Pascal so indeksi od 1, kar pomeni, da je zaporedje indeksov 4, 2, 1 in 3):

90	64	50	78
3	1	0	2

Očitno lahko števila iz *prve vrstice razpredelnice* zelo enostavno uredimo po naraščajočem vrstnem redu, če poznamo števila iz *druge vrstice razpredelnice*. Drugi vrstici razpredelnice pravimo tudi *permutacijski vektor*.

6.1. Navedenim številom v drugo vrstico razpredelnice pripišite permutacijski vektor:

10	61	74	30	27	47	73	70	82	5

(3 točke)

6.2. Imamo tabelo 2015 števil. Opišite postopek ali napišite funkcijo `urejena(stevila)`, ki preveri, ali so števila v tabeli `stevila` urejena naraščajoče: vrne `1` oziroma `True`, če so števila pravilno urejena, sicer pa `0` oziroma `False`. Pozor, v tabeli `stevila` so samo števila in ne permutacijski vektor – se pravi, to je prva vrstica iz naše razpredelnice.

(4 točke)

6.3. Če znamo naključno tvoriti permutacijske vektorje, lahko zelo preprosto uredimo števila (prva vrstica iz razpredelnice) z algoritmom *bogosort*:

```
def bogosort(stevila):
    while not urejena(stevila):
        v = permutacijski_vektor(len(stevila))
        stevila = permutiraj(stevila, v)
```

Funkcijo `urejena` smo naredili v nalogi 6.2. Recimo, da nam je Peter Zmeda pripravil funkcijo `permutacijski_vektor(dolzina)`, ki vrne naključni permutacijski vektor dolžine `dolzina` (z drugimi besedami, vrne nam drugo vrstico naše razpredelnice). Zato, da napišemo program `bogosort`, nam manjka samo še funkcija `permutiraj`.

Opišite postopek ali napišite funkcijo `permutiraj(stevila, v)`, ki števila v tabeli `stevila` (prva vrstica iz razpredelnice) velikosti 2015 prerazporedi (permutira), kot zahteva permutacijski vektor `v` (druga vrstica iz razpredelnice). Postopek oziroma funkcija naj vrne prerazporejeno tabelo števil.

Primer: recimo, da je tabela števil 90, 64, 50 in 78 ter je permutacijski vektor 2, 1, 0 in 3, potem je rezultat funkcije tabela števil 50, 64, 90 in 78:

`permutiraj([90,64,50,78], [2,1,0,3])` vrne tabelo `[50,64,90,78]`

(3 točke)

M 1 6 2 4 5 1 1 2 1 4

V sivo polje ne pišite.

Prazna stran

V sivo polje ne pišite.

Prazna stran

M 1 6 2 4 5 1 1 2 1 6

V sivo polje ne pišite.

Prazna stran