

Testiranje programov s pomočjo programa `tj.exe`

V tem dokumentu podajamo navodila za uporabo programa `tj.exe` (razdelek 1). Ker pa boste vaše programe gotovo kdaj testirali tudi ročno, ta način preverjanja programov opisujemo v razdelku 2.

1 Avtomatsko testiranje s pomočjo programa `tj.exe`

Program `tj.exe` je bil prvotno mišljen kot različica enostavne bashevske skripte za MS okna, nato pa ga je kolega Poženel tako dodelal, da smo ga promovirali v uradni program za testiranje vaših rešitev domačih in izpitnih nalog. Program odlično deluje tudi na linuxu; potrebujemo le ogrodje `mono` (<http://www.mono-project.com>), ki ga na sistemu ubuntu namestimo tako:

```
sudo apt-get install mono-complete
```

(Če nam primanjkuje prostora, bi najbrž zadoščalo namestiti paket `mono-runtime` in po potrebi še kakšno knjižnico `libmono-*`, a tega nismo preverjali.) Mono je možno namestiti tudi na jabolku, zato bi `tj.exe` moral delovati tudi tam.

Naj omenimo, da je program `tj.exe` na virtualki *že* nameščen.

1.1 Osnovna uporaba

Uporaba programa `tj.exe` je odvisna od načina testiranja:

- Pri načinu testiranja **vhod-izhod** (torej z množico parov datotek `vhod*.txt` in `izhod*.txt`) program poženemo takole:

```
tj.exe <Program.java> <testi> <rezultati>
```

Pri tem je `<Program.java>` javanski program, ki ga želimo testirati, `<testi>` je mapa, ki vsebuje datoteke `vhod*.txt` in `izhod*.txt`, `<rezultati>` pa je mapa, v katero naj program izpiše rezultate preverjanja in poročilo v obliki HTML (`prikaz.htm`). **Znakov `< in >` ne pišemo!**

Na izpitu se bo vse troje nahajalo v isti mapi, zato boste program znotraj te mape pognali takole:

```
tj.exe <Program.java> . .
```

Za `<Program.java>` sledijo presledek, pika, presledek in pika.

Program `tj.exe` primerja pričakovane izhode (`izhod*.txt`) z izhodi, ki jih na podlagi posameznih vhodov proizvede testirani program `<Program.java>`. Dobljeni izhodi se zapišejo v datoteke `rezultat*.txt` v mapo `<rezultati>`. Če

namesto mape `<rezultati>` uporabimo piko, se bodo datoteke `rezultat*.txt` (in tudi poročilo `prikaz.htm`) zapisale v trenutno mapo.

- Pri testiranju s **testnimi razredi** (torej z množico parov datotek `Test*.java` in `izhod*.txt`) program poženemo takole:

```
tj.exe <razredi> <testi> <rezultati>
```

Pri tem `<razredi>` označuje mapo, ki vsebuje javanske datoteke, ki jih želimo testirati, `<testi>` je mapa, ki vsebuje datoteke `Test*.java` in `izhod*.txt`, `<rezultati>` pa je mapa, v katero naj program izpiše rezultate preverjanja in poročilo v obliki HTML (`prikaz.htm`).

Če se vse troje nahaja v isti mapi, potem lahko program znotraj te mape poženemo kar brez argumentov:

```
tj.exe
```

Program `tj.exe` primerja pričakovane izhode (`izhod*.txt`) z izhodi, ki jih na podlagi testiranih javanskih datotek v mapi `<razredi>` proizvedejo posamezni testni razredi v mapi `<testi>`. Dobljeni izhodi se zapišejo v datoteke `rezultat*.txt` v mapo `<rezultati>`. Če program `tj.exe` poženemo brez argumentov, se bodo datoteke `rezultat*.txt` (in tudi poročilo `prikaz.htm`) zapisale v trenutno mapo.

- Pri testiranju **grafike** so pravila povsem enaka kot pri testiranju s testnimi razredi, saj se grafika vedno preverja na ta način. Če se javanske datoteke ter testni razredi in pripadajoče izhodne slike nahajajo v isti mapi, potem program `tj.exe` znotraj te mape poženemo brez argumentov:

```
tj.exe
```

Program `tj.exe` preverja podobnost posameznih pričakovanih slik (`izhod*.png`) s slikami, narisanimi na podlagi testnih razredov. Te se shranijo v datoteke `rezultat*.png` v mapi `<rezultati>` oziroma v trenutni mapi, če program `tj.exe` poženemo brez argumentov.

Kako program `tj.exe` ločuje med »navadnimi« testnimi razredi in testnimi razredi za preverjanje grafike? Na vrhu vsakega »grafičnega« testnega razreda se nahaja vrstica

```
// _NE_ODSTRANI_
```

ki programu `tj.exe` pove, da bo moral primerjati par slik namesto dveh kosov besedila. Če to vrstico odstranimo, program seveda ne bo deloval!

1.2 Poročilo prikaz.htm

Program `tj.exe` po vsakem zagonu proizvede poročilo `prikaz.htm`. Poročilo si lahko ogledate v brskalniku. Z brskalnikom si namreč lahko ogledujemo poljubne datoteke HTML, ne samo spletne strani. Datoteko `prikaz.htm` si ogledamo tako, da jo v brskalniku odpremo s pomočjo ukaza `File/Open`.

Priporočamo vam, da imate okno brskalnika nenehno odprto in da ga po vsakem zagonu programa `tj.exe` osvežite s kombinacijo `Ctrl-R`. Poročilo prikazuje posamezne vhode (oziroma testne razrede) in pripadajoče izhode, v primeru neskladja med dobljenim in

pričakovanim izhodom pa prikaže oba. Če testiramo grafiko, prikaže dobljene in pričakovane slike za posamezne testne primere. Če si sliko želite ogledati v polni ločljivosti, nanjo desno kliknite in izberite **View Image** ali kaj podobnega.

1.3 Dodatni parametri programa `tj.exe`

Program `tj.exe` podpira tudi uporabo dodatnih stikal; za ogled vseh poženite

```
tj.exe -?
```

S stikalom `-t` lahko nastavimo časovno omejitev (privzeta je 1 sekunda). Na primer, z ukazom

```
tj.exe -t 5s <morebitni_ostali_parametri>
```

podaljšamo časovno omejitev na 5 sekund, z ukazom

```
tj.exe -t 100ms <morebitni_ostali_parametri>
```

pa jo skrajšamo na 100 milisekund.

Najbrž pa vam bo še precej bolj koristilo stikalo `-p`, s katerim lahko testiramo samo izbrane primere. Na primer, z ukazom

```
tj.exe -p 6-9 <morebitni_ostali_parametri>
```

testiramo samo primere s števkami od 6 do vključno 9 (torej `Test06.java` do vključno `Test09.java` v primeru testiranja s testnimi razredi). Poročilo `prikaz.htm` se bo tvorilo samo za izbrane primere.

Če kot parameter stikala `-p` navedemo samo en testni primer, potem se poročilo `prikaz.htm` ne bo tvorilo, saj se bodo podrobni rezultati testiranja primera prikazali kar v terminalu. Na primer, sledeči ukaz izpiše podrobne rezultate testiranja primera št. 7:

```
tj.exe -p 7 <morebitni_ostali_parametri>
```

Če želite testirati en sam primer in rezultate zanj kljub vsemu izpisati v poročilu `prikaz.htm`, uporabite sledečo obliko:

```
tj.exe -p 7-7 <morebitni_ostali_parametri>
```

2 Ročno testiranje

Čeprav poročilo `prikaz.htm` prikazuje tudi izjeme (napake med izvajanjem vašega programa), vam bo morda vseeno kdaj prišlo prav tudi ročno preverjanje programa. V ta namen moramo program (in testne razrede) seveda najprej ročno prevesti, nato pa pognati. Če želimo vhod prebrati iz datoteke namesto iz tipkovnice, izhod pa izpisati v datoteko namesto na zaslon, lahko to storimo, recimo, takole:

```
java Program <vhod07.txt> rezultat07.txt
```

oziroma

```
java Test07 > rezultat07.txt
```

Datoteko `rezultat07.txt` nato primerjamo z datoteko `izhod07.txt`. Lahko obe odpremo v urejevalniku, lahko pa se poslužimo programa `diff` (linux / jabolko) oziroma `fc` (okna).

Pri testiranju grafike lahko ročno poženemo izbrani testni razred (razred `TestK.java` bo izdelal datoteko `rezultatK.java`), lahko pa poženemo tudi program, ki smo ga napisali. Če ga poženemo brez parametrov (`java Program`), se bo odprlo grafično okno s sliko, ki jo program nariše. Lahko pa ga poženemo tudi s parametri, na primer takole:

```
java Program rezultat.png 700x500
```

Slika se bo shranila v datoteko `rezultat.png`, in sicer v ločljivosti 700×500 slikovnih elementov (»pikslov«).