

**Slika 31:** Podobni strukturi se odražata tudi v podobnih zapisih ukazov. Potrebni sta pozornost in natančnost.

Ukaza sta naslednja:

```
if (pogoj1){ if (pogoj2){ukazi1} else {ukazi2}}
if (pogoj1){ if (pogoj2){ukazi1}} else {ukazi3}
```

Ukaza sta na videz zelo podobna. Tu je vredno še enkrat ponoviti recept: z zamikanjem, pisanjem v več vrsticah in z dodatnim oklepaji lahko programsko kodo naredimo bolj pregledno in nedvoumno. Morda takole:

```
if (pogoj1)
  {if (pogoj2){ukazi1}
   else {ukazi2} }

if (pogoj1){ if (pogoj2) {ukazi2} }
else {ukazi3}
```

### 5.3.2 Veriženje pogojev : *else if*

Več pogojev lahko "verižimo" s kombinacijo *else if*. Uporabnost tega načina si oglejmo na primeru:

*Dan je kriterij za ocenjevanje. Število točk je treba pretvoriti v besedo oceno po naslednjem kriteriju:*

- Za 90 ali več točk je ocena "odlično".*
- Za 78 ali več točk je ocena "prav dobro".*
- Za 65 ali več točk je ocena "dobro".*
- Za 50 ali več točk je ocena "zadostno".*
- Za manj kot 50 točk je ocena "nezadostno".*

Naloga poznamo in jo razumemo. Vendar pa je kriterij napisan nepopolno. Kriterij za oceno "prav dobro" namreč velja, če ni bil izpolnjen kriterij (število točk) za oceno "odlično"<sup>1</sup>. Podobno velja za oceno "dobro" le , če pred tem ni bilo dovolj točk za oceno "prav dobro"...

<sup>1</sup> Sicer bi bil s 100 točkami izpolnjen pogoj za ocene od "zadostno" do "prav dobro".

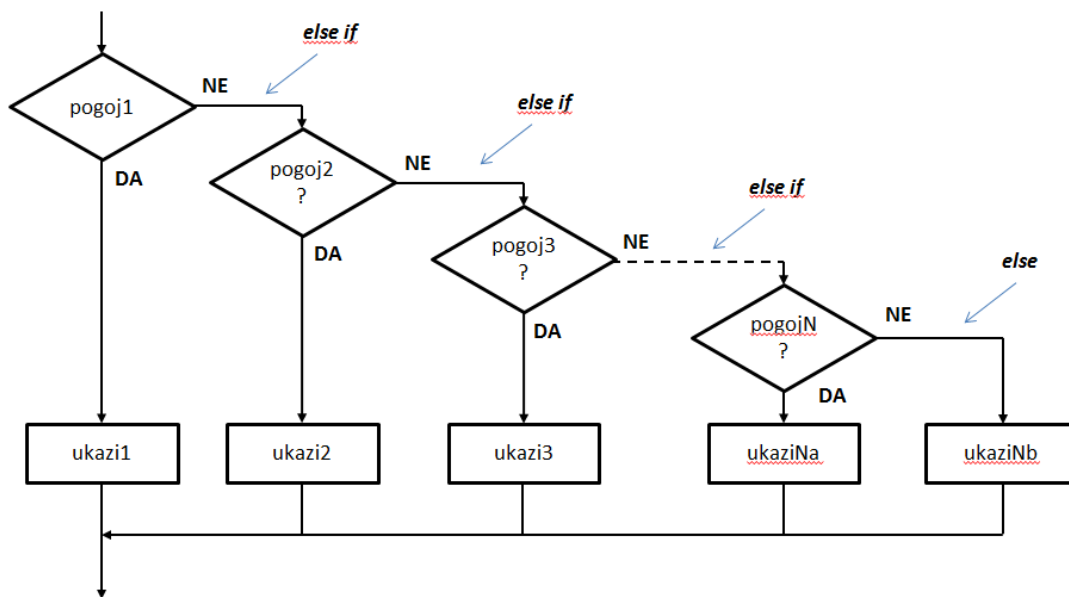
Nalogo bi lahko rešili s pogojnimi stavki. Za oceno "dobro" bi se pogojni stavek glasil:

```
if (tocke >= 65 && tocke < 78) {ocena = "dobro"};
```

Slabost tega načina je, da pogoji v stavkih niso med seboj 'povezani', čeprav v resničnosti so. Možnost napak je večja. Če bi v gornjem primeru namesto  $\geq 65$  pomotoma napisali samo  $> 65$  bi za 65 točk ne imeli pripadajoče ocene.

Bližje realnosti bi bilo *gnezdenje pogojev*, kjer bi dobili precej razvejano strukturo pogojnih stavkov.

Java pa omogoča uporabo kombinacije *else if*, ki ne zahteva zaključevanja za nazaj. Diagram poteka za takšno strukturo prikazuje slika.



**Slika 32:** Lestvična struktura diagrama poteka z uporabo *else if*. Na zadnjem izhodu z NE je samo *else*.

Z uporabo *else if* napišemo rešitev za našo nalogo takole:

```
if (tocke > 90) {ocena = "odlično";}
else if (tocke > 78) {ocena = "prav dobro";}
else if (tocke > 65) {ocena = "dobro";}
else if (tocke > 50) {ocena = "zadostno";}
else {ocena = "nezadostno"};
```

Stavek napišemo skoraj natanko tako, kot razmišljamo in govorimo v vsakdanjem življenju:

*Če nisem pisal(a) pet, sem morda vsaj štiri?*

*Če nisem pisal(a) niti štiri, sem morda vsaj tri?*

*Če nisem pisal(a) niti tri, sem morda vsaj dve?*

*Če nisem pisal(a) niti dve, (potem nimam več podvprašanj in) sem pisal(a) ... (nee! ☹).*

Takšno veriženje pogojev je pregledno in učinkovito. Poleg tega zadnji *else* 'pospravi' vse vrednosti, ki jih nismo zajeli s predhodnimi pogoji. Prepičani smo lahko, da nobena vrednost ni spregledana. Seveda pa moramo vrednosti za številske pogoje napisati po velikosti in v pravi smeri.