

PRIPRAVA PROGRAMA

Dobro smo se že seznanili s strojnimi jeziki - orodjem za izdelavo naših programov. A to ni vse! Tudi za gradnjo ni dovolj, da poznamo orodje in znamo zlagati opeke. Hiša, zidana brez načrta bo hladna, grda in se bo kmalu podrla. Enako je s programi! Nikoli ne smemo graditi brez načrta. Kako torej zastaviti delo, da bo program dober, pravilen in razumljiv?

Pisanje programov v besicu vam je gotovo domače. Programe pogosto vnašamo kar "iz glave", a so navadno kljub temu dovolj uspešni. Nadzorni program nas opozori na napake, dolžina programov pa je navadno tolikšna, da so še pregledni in hitro razumljivi. V strojnem jeziku je drugače. O napakah ni nobenih obvestil, spodrsljaji se končajo slabo. Programi so dolgi in (brez komentarja) zelo težko razumljivi. Zaradi tega priprava strojnih programov ne sme biti le "pisanje programov", temveč pravo programiranje. A kaj pravzaprav je programiranje? To je zbirka opravil, ki jih navadno razdelimo v tri stopnje:

- načrtovanje programa in logično preizkušanje načrta,
- kodiranje ter
- preizkušanje in popravljanje.

Najpomembnejša in časovno najdaljša je prva stopnja. Kolikor bolje je izdelan načrt, toliko manj dela in časa porabimo kasneje za ostala opravila.

NACRTOVANJE

Program je nemogoče oblikovati naenkrat. Tega pravila ne smemo pozabiti. Marsikdo se loti izdelave strojnega programa kar "na pamet". Tako nastajajo programi s številnimi spodrsljaji, ki jih nato množica popravkov in dopolnitev spremeni v nepregledno gmoto ukazov. Zato ne moremo dovolj toplo priporočiti vestne priprave načrta na način, ki ga imenujemo *od vrha navzdol*. Nalogo (naj bo igra ali uporabni program), ki jo moramo rešiti, razdelimo na več manjših korakov. Zaporedje vseh korakov, ki vodijo do rešitve, imenujemo algoritem.

Denimo, da bi želeli izdelati program vožnje na avtomobilski dirki. Algoritem bi na prvi mah izgledal nekako tako:

Čakaj Če ni pritisnjena nobena tipka, pojdi na Čakaj.
Vožnja Premakni avto.

Preveri smer in hitrost.

Če smer ali hitrost nista pravi, pojdi na Trk.

Če vozilo še ni v cilju, pojdi na Vožnja.

Izpiši doseženi čas in uvrstitev.

Pojdi na Čakaj.

Izpiši, da se je zgodila nesreča.

Pojdi na Čakaj.

Kot vidite, so vsa navodila zapisana v navadni slovenščini. To je zlato pravilo: algoritem naj ostane čim dlje zapisan v običajnem jeziku. Nismo se še odločili, ali bo program v basicu, strojnem ali kakšnem drugem jeziku. Takšna odločitev bi nas trenutno le ovirala - zamisel in načrt programa nista odvisna od jezika, v katerem bomo zapisali program.

Ko imamo korake zapisane, jih še enkrat pregledamo. Smo kaj pozabili? Je kakšen korak odveč? Katere korake lahko oblikujemo kot podprograme? Tak pregled imenujemo tudi logično preizkušanje. Preglejmo naš program! Zanimarili smo navodila zato dodajmo na začetku korak, ki jih bo napisal. Če dobro pogledamo, ni nikjer koraka, v katerem bi končali program. Dodajmo torej korak **Konec**. Zaradi tega moramo spremeniti še zadnja dva koraka:

Cilj Izpiši doseženi čas in uvrstitev.

Pojdi na Konec.

Trk Izpiši, da se je zgodila nesreča.

Pojdi na Konec.

Konec Vprašaj, ali želi igralec končati igro.

Če ne želi, pojdi na Čakaj.

Končaj.

Z algoritmom smo zdaj zadovoljni in lahko nadaljujemo z različnitvijo posameznih korakov. **Konec** bi natančneje zapisali:

Konec

Obriši zaslon.

Izpiši "Ali želiš končati?"

Preleti tipkovnico.

Če je pritisnjen "d", končaj.

Pojdi na Čakaj.