Razvoj programske opreme z jezikom SDL

1 Uvod

SDL je formalni jezik, ki je razširjen predvsem na področju telekomunikacij, za kar je bil tudi razvit. Specifikacije v SDL-ju opišemu grafično ali tekstovno. Pri vajah bomo uporabljali grafične gradnike povsod, kjer bo le mogoče.

Ločimo statični in dinamični opis sistema. Statični opis začnemo s simbolom za sistem. Sistem razdelimo v množico blokov, ki so organizirani hierarhično po nivojih. Bloki med seboj komunicirajo s signali po kanalih. Signali lahko nosijo sporočila.

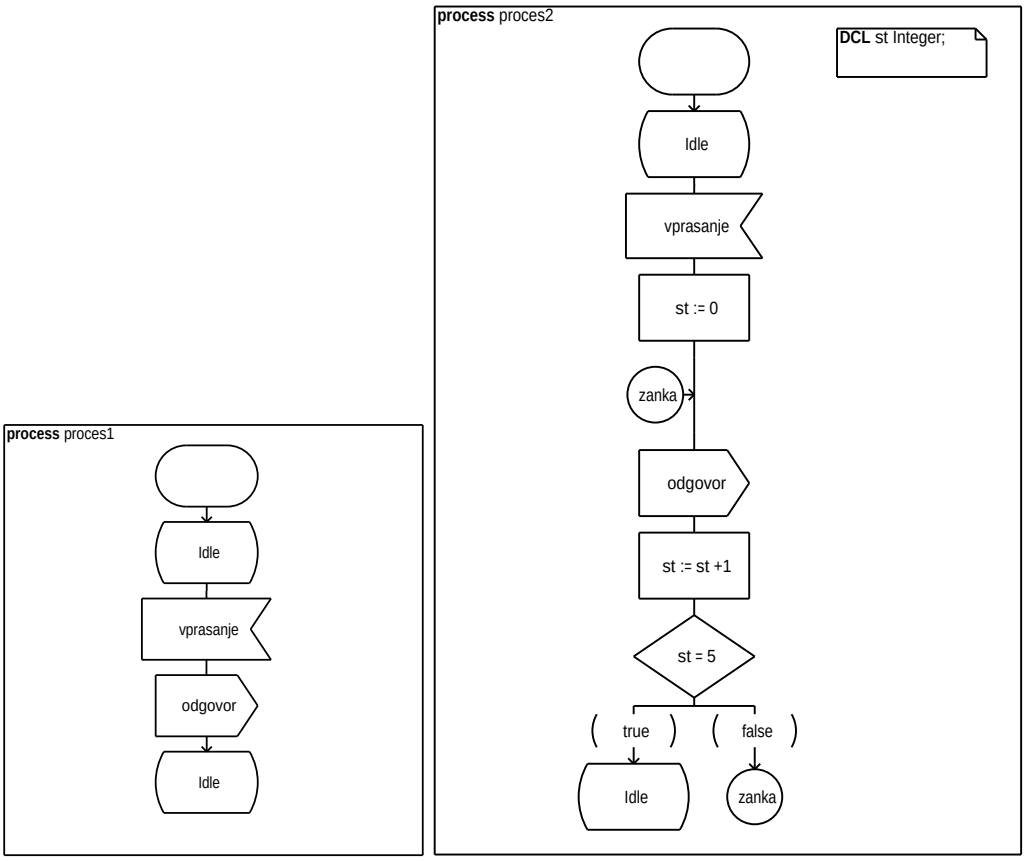
Narišite gradnike statičnega opisa sistema:

Vsak blok, ki je list strukture, lahko vsebuje procese. Procesi so razširjenimi končni avtomati, ki predstavljajo dinamični opis sistema. Vsak proces se vedno nahaja v nekem stanju. Znotraj enega bloka procesi med seboj komunicirajo s signali po signalnih poteh. Po prejemu določenega signala proces izvede zaporedje podanih akcij in preide v novo stanje. Najpomembnejši gradniki, s katerimi opišemo obnašanje procesov, so: *start*, *state*, *input*, *output*, *save*, *decision* in *task*.

Narišite gradnike dinamičnega opisa sistema:

1.1 Zanke

Pri specifikaciji procesov pogosto naletimo na potrebo, da naredimo zanko. SDL nam omogoča dve vrsti zank. Najenostavneje je, če proces načrtujemo tako, da se iz trenutnega stanja vrne nazaj v stanje, v katerem je že bil (slika 1). Slabost takšne zanke je v tem, da se lahko vrnemo le v poimenovano stanje. Če se želimo vrniti na kakšno drugo mesto, uporabimo gradnika *label* in *join*. V drugem primeru na sliki 1 se bo tako za vsak dobljen signal *vprasanje* oddalo pet signalov *odgovor*.

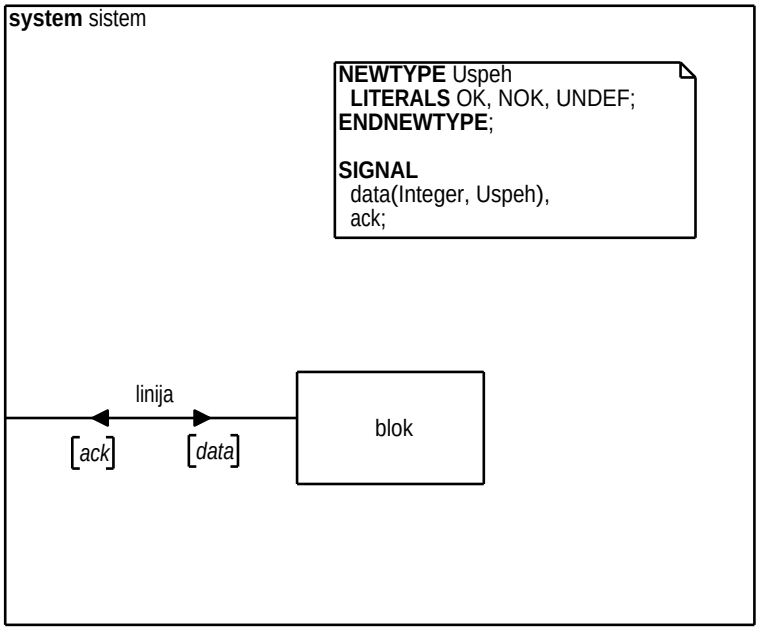


Slika 1: Specifikacija zanke

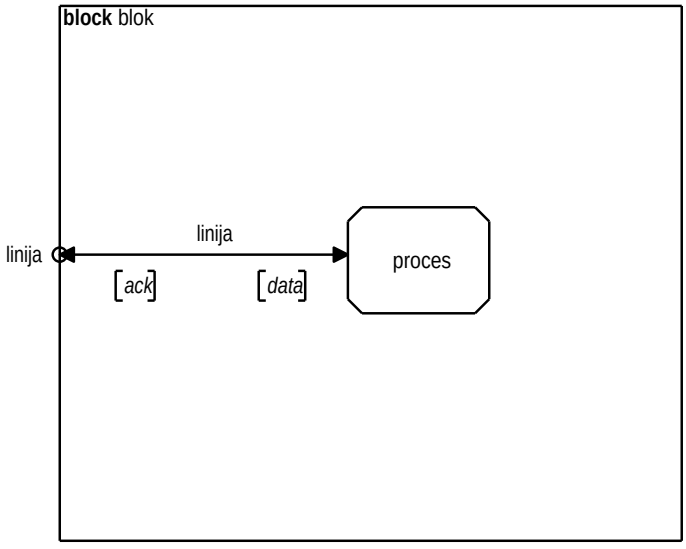
1.2 Signali in parametri

Procesi med seboj komunicirajo s pomočjo signalov. Signal ima v osnovni obliki samo ime. Če želimo, da prenaša tudi kakšna sporočila, mu dodamo parametre. Za vsak parameter moramo podati tip (npr. *Integer* za cela števila). Pogosto potrebujemo parametre, ki lahko zavzame le nekaj naštetih vrednosti. Takšne naštevne tipe deklariramo z ukazom *NEWTYPE*.

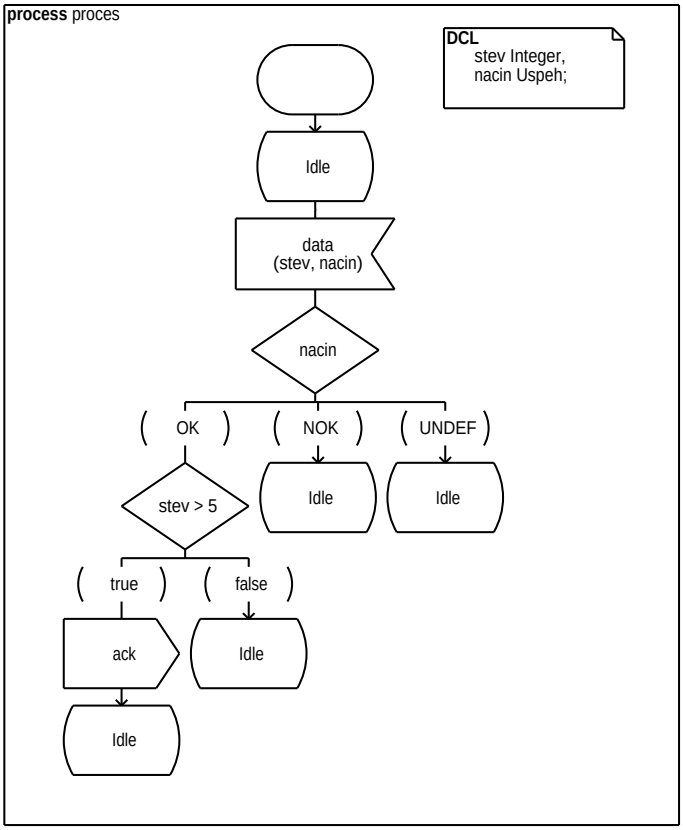
Primer uporabe parametrov je na slikah 2, 3 in 4. Proces bo oddal signal *ack* le, če bosta parametra signala *data* ustrezna: *stev* mora biti večji od 5 in *nacin* mora biti enak *OK*.



Slika 2: Uporaba parametrov



Slika 3: Uporaba parametrov - nadaljevanje



Slika 4: Uporaba parametrov - nadaljevanje

Tukaj si zabeležite opombe v zvezi z uporabo parametrov:

1.3 Janko in Metka

Janko in Metka sta vsak s svojim telefonskim aparatom povezana na isto krajevno telefonsko centralo. Zaradi preprostosti vzemimo, da sta edina povezana na to centralo in da občasno (v pravem trenutku) pokliče Janko Metko, Metka pa Janka nikoli. Metka občasno telefonira prijateljicam in takrat je njena povezava zasedena.

Vemo, da se ob dvigu slušalke centrali avtomatično pošlje signal off-hook. Centrala si označi pripadajoči aparat kot zaseden, odgovori s tonskim signalom dial tone ter čaka na odgovor. Kadar Janko sliši dial tone po dvigu svoje slušalke, običajno takoj vtipka klicno številko. Metkino številko si izberite sami. Vzemite, da se Janko včasih zmoti in namesto Metkine številke vtipka kakšno drugo. Vtipkana številka se v vsakem primeru v naslovnem signalu prenese k centrali.

Včasih pa se Janko po prejemu dial tone premisli in položi slušalko. Vemo, da se ob položitvi slušalke centrali avtomatično pošlje signal on-hook. Centrala si pripadajoči aparat zapomni kot prost in je pripravljena na nov klic z njega.

Ko centrala prejme številko klicanega, najprej preveri, ali je prava. Če ni, pošlje kličočemu sporočilo o napačni številki(wrong). Ko kličoči dobi to sporočilo, odloži slušalko. Druga možnost je, da centrala prejme Metkino številko. V tem primeru preveri, ali je zasedena. Če je, pošlje Janku slišni signal busy. Če Metkina številka ni zasedena, ji prične pošiljati signal zvonjenja, podobnega pa tudi Janku.

Če Metka ne dvigne slušalke, centrala po določenem času pošlje Janku signal disconnected. Če Metka dvigne slušalko, centrala prejme od nje signal off-hook in takoj pošlje Janku signal connected, da je zveza vzpostavljena. Centrala je sedaj v stanju, ko se Janko in Metka pogovarjata. V tem stanju ostane, dokler eden od njiju ne odloži slušalke. Če dobi on-hook od Janka, pošlje signal disconnected Metki, in obratno.

Vzemimo, da Janko oz. Metka po prejemu signala busy oz. disconnected vedno odložita slušalko. Medtem, ko Janko prejema znak, da pri Metki zvoni, lahko odloži slušalko in centrali se pošlje on-hook. Signal zvonjenja pri Metki preneha.

Naš cilj je z jezikom SDL opisati krajevno telefonsko centralo kot reaktivni sistem, ki sprejema signale iz okolja, tj. od Jankovega in Metkinega telefonskega aparata, in pošilja odzive nazaj v okolje. Na vajah boste dobili opis obnašanja Jankovega in Metkinega telefona, sami pa boste sestavili opis centrale. Opis centrale preizkusite z značilnimi zaporedji signalov od Janka in Metke.