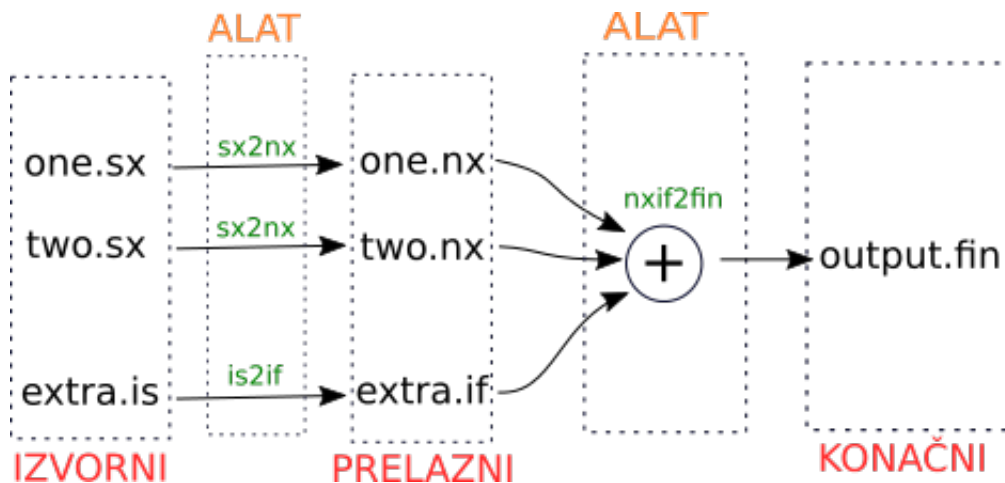


VEŽBA 3

Pregled

Prevođenje složenijih softverskih sistema iz oblika izvornog programskog koda (source) u konačni izvršni oblik (engl. *executable*) je obično složen proces. Uvodi se čitav niz zavisnosti u sistem koji se dodatno komplikuje postojanjem niza prelaznih fajlova. Celokupan proces je ilustrovan na sledećoj slici gde strelica pokazuje međusobnu zavisnot fajlova – strelica pokazuje od kojeg fajla zavisi koji fajl. Jedan fajl često može zavisiti od niza fajlova što je na slici ilustrovano simbolom sabirača.



Savremeni sistemi za razvoj softvera redovno sadrže alat koji automatizovano razrešava međusobne zavisnosti fajlova uključene u sistem. Alat se zove *make*, a međusobne zavisnosti fajlova opisane su u odgovarajućem fajlu koji nosi ime *makefile* i neophodno je da se nalazi u radnom direktorijumu u trenutku poziva make alata.

Zadaci

1. Preuzeti gotov *makefile* i kopirati ga u pripremljeni radni direktorijum. Pripremiti odgovarajuće izvorne fajlove *one.sx*, *two.sx* i *extra.is* (u skladu sa gornjom šemom). Pokrenuti alat *make* u *dry-run* modu komandom *make -n*. Ovaj oblik komande samo ispisuje na terminalu koje će komande pozivati, ali ih ne poziva zapravo. Proučiti ispisani niz komandi.
2. Napisati skript *is2if* koji će imati ulogu alata za prevođenje izvornih fajlova *.is* u prelazni *.if* oblik. Nakon testiranja minimalnim izmenama ovog alata kreirati i alat za prevođenje *.sx* u *.nx* oblik – alat *sx2nx*.

Alat treba da se poziva u formi *is2if -i <input_is_file> -o <output_if_file>*. Neophodno je obezbediti obradu parametara unutar skripta kao i proveru njihove ispravnosti: da li su uopšte zadati *input_is_file* i *output_if_file*, i postoji li *input_file*. Ukoliko se otkrije bilo koja od navedene tri greške obustaviti rad skripta i vratiti status greške (1, 2 i 3 redom) uz izdavanje poruke o grešci na kanalu *stderr*.

Alat *sx2nx* treba da bude isti kao i *is2if*, izmene treba uraditi samo na mestima gde se alat predstavlja korisniku – u porukama. Predstavljanje programa je veoma važno u procesu otkrivanja grešaka pri korišćenju alata.

Sadržaj izlaznog fajla treba da bude:

Kreiran na osnovu fajla `<input_file>`, zapisan u fajl `<output_file>`.

`<input_file>` i `<output_file>` se zamenjuju pravim imenima fajlova.

Primer (ispravan rad):

```
> is2if -i extra.is -o extra.if
is2if: Obradjujem extra.is i zapisujem rezultat u extra.if
>
```

Primer (greška – ulazni fajl ne postoji):

```
> is2if -i emtrax.is -o extra.if
is2if: Ulazni fajl pod imenom emtrax.is nije pronađen!
Obustavljam.
>
```

Primer (greška – nisu zadati svi parametri):

```
> sx2nx -i two.sx
sx2nx: Nije zadat izlazni fajl!
Obustavljam.
>
```

3. Napisati skript *nxif2fin* koji će imati ulogu alata za kreiranje konačnog oblika (*.fin*) na osnovu niza *.nx* i *.if* fajlova.

Alat treba da se poziva u formi *nxif2fin <infile_1> <infile_2> <infile_3> -o <outfile> ... <infile_n>*. Obratiti pažnju da je moguće zadati proizvoljan broj ulaznih fajlova i jedan izlazni fajl. Izlazni fajl je obeležen prekidačem *-o* neposredno ispred navođenja imena fajla. Ovo reflektuje činjenicu da ovaj alat spaja više ulaznih fajlova u jedan.

Neophodno je obezbediti obradu parametara unutar skripta kao i proveru njihove ispravnosti: da li je uopšte zadat ijedan *infile*, a ako jeste postoje li oni? Treba proveriti i da li je zadat *outfile*. Ukoliko se otkrije bilo koja od navedene tri greške obustaviti rad skripta i vratiti status greške po slobodnom izboru uz izdavanje poruke o grešci na kanalu *stderr*.

Kao i kod prethodnih alata, i *nxif2fin* treba da se predstavi u svakoj svojoj poruci, čak i kada greške nema onako kako je prikazano u primerima.

Sadržaj izlaznog fajla treba da bude:

Kreiran na osnovu fajla (fajlova) `<spisak infile stavki>`, zapisan u fajl `<output_file>`.

Kao i u prethodnom slučaju `<spisak infile stavki>` i `<output_file>` se zamenjuju pravim imenima zadatih fajlova.

U rešenju koristiti sledeću tehniku izgradnje liste:

```
list="$list $novi_element"
```

Navedena linija se postavlja unutar petlje, a u svakom prolasku promenljiva `novi_element` sadrži stavku koju treba dodati na listu.

Primer (ispravan rad):

```
> nxif2fin one.nx two.nx extra.if -o output.fin
nxif2fin: Obradjujem one.nx two.nx extra.if extra.is
zapisujem rezultat u output.fin
>
```

Primer (greška – ne postoji jedan od ulaznih fajlova):

```
> nxif2fin onex.nx two.nx extra.if -o output.fin
nxif2fin: Ulazni fajl onex.nx ne postoji!
Obustavljam.
>
```

4. Testirati kompletan sistem pomoću *make* alata u redovnom režimu rada. Pratiti poruke koje se ispisuju na ekranu. Eksperimentisati sa izmenama pojedinih fajlova i pratiti kako će se to odraziti na sledeći ciklus obrade.