

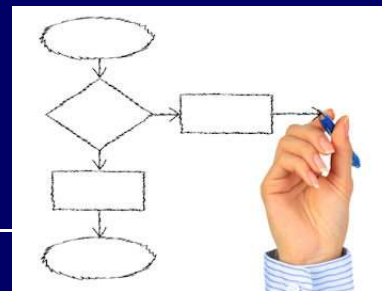
POJAM ALGORITMA

- **Algoritam** je precizno opisan postupak za rešavanje nekog problema.
- Obično je to spisak uputstava ili skup pravila kojima je, korak po korak, opisan postupak za rešavanje zadatog problema.
- Svaki korak algoritma, odnosno svako uputstvo iz spiska, mora da bude definisana operacija.



POJAM ALGORITMA

- Algoritmi u računarstvu imaju veoma značajnu ulogu.
- Postoji i posebna naučna disciplina koja se bavi problematikom algoritama - **teorija algoritama.**
- Algoritmi moraju da budu nedvosmisleni i da završavaju u konačnom broju koraka.





FAZE REŠAVANJA ZADATKA PRIMENOM RAČUNARA

1. Analiza, definicija i precizna identifikacija problema.
2. Globalni opis odgovarajućeg algoritma,
3. Detaljna izrada algoritma.
4. Pisanje programa za računar na osnovu detaljnog algoritma.
5. Testiranje programa na računaru.
6. Izvođenje programa za rešavanje datog zadatka.



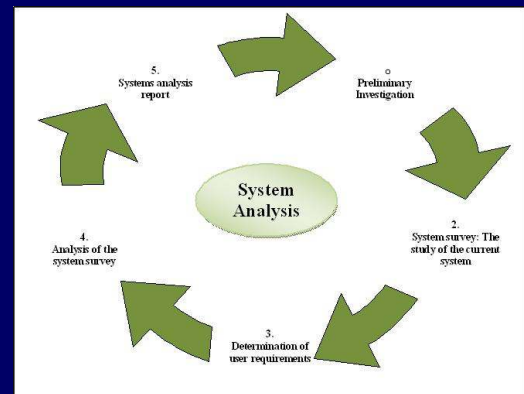
SISTEMSKA ANALIZA

- Problematikom što preciznijeg definisanja problema koje treba rešiti primenom računara bavi se posebna grana računarstva koja se naziva **sistemska analiza**.



ANALIZA, DEFINICIJA I PRECIZNA IDENTIFIKACIJA PROBLEMA

- Analiza, definicija i precizna identifikacija problema je veoma bitna faza programiranja, mada joj se često ne posvećuje dovoljna pažnja.



GLOBALNI OPIS I DETALJNA IZRADA ALGORITMA

- Kada je dovoljno precizno obavljena prva faza i kada su jasno sagledani zahtevi prema računaru u rešavanju konkretnog problema pristupa se globalnom opisu algoritma na osnovu čega se kasnije detaljno izrađuje odovarajući algoritam.



PISANJE PROGRAMA

- Sistematičnost i postupnost su veoma važni kod izrade algoritma kako bi kasnije pisanje programa bilo što jednostavnije.



TESTIRANJE PROGRAMA

- Kada se program napiše i unese u računar sledi faza testiranja programa na računaru.
- U toku ove faze uočavaju se i otklanjaju eventualne greške ili pojedina loša programska rešenja.



TESTIRANJE PROGRAMA

- Ova faza može da bude i dugotrajna, pa čak i da nas vrati na prvu fazu ukoliko se pokaže da još u pripremnim radnjama nismo dovoljno dobro analizirali problem koji treba da rešimo.



IZVOĐENJE PROGRAMA ZA REŠAVANJE DATOG ZADATKA

- Program može da se koristi sa pravim podacima i za rešavanje praktičnih problema zbog kojih je i napravljen tek kada se u fazi testiranja uverimo da je pouzdan.
- Prilikom izrade algoritama krajnji cilj je uvek izrada programa za računar.



BLOK DIJAGRAMI

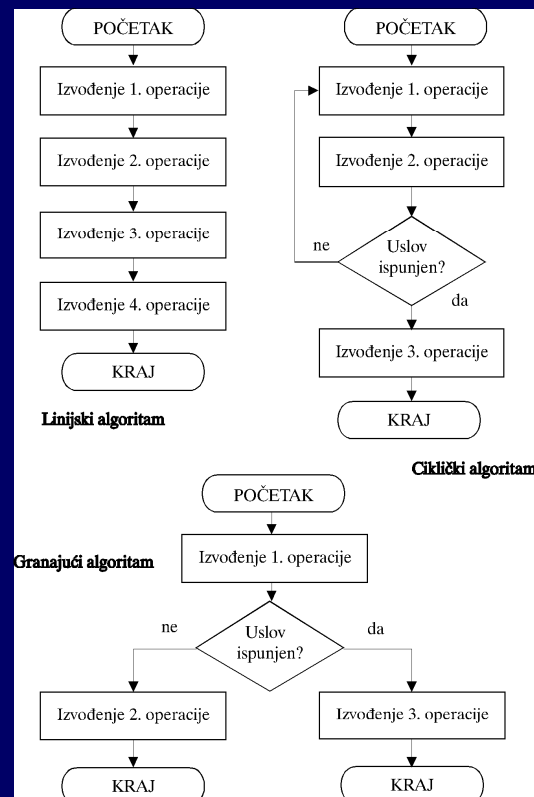
- Program, u stvari, predstavlja opis algoritma u nekom programskom jeziku.
- Radi preglednosti često se algoritmi predstavljaju posebnim grafičkim simbolima u obliku posebnog grafičkog prikaza poznatog pod imenom **blok-dijagram**.



ALGORITMI

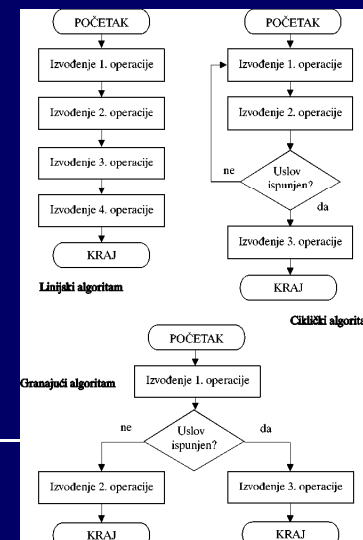
Prema načinu izvršavanja instrukcija algoritmi mogu da budu:

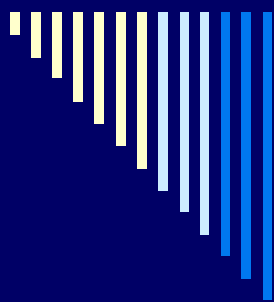
- linearni,
- ciklički,
- granajući.



ALGORITMI

- U linearnim algoritmima se svaka operacija izvršava samo jedanput.
- U cikličnim se neke operacije izvršavaju više puta.
- U granajućim se neke operacije izvršavaju u zavisnosti od ispunjenosti određenog uslova.





PROGRAMI, PROGRAMIRANJE I PROGRAMSKI JEZICI

- Programi su algoritmi opisani na način koji može da razume računar.
- Program je, dakle, niz instrukcija koje nalažu računaru šta da uradi i kako to da obavi.
- Čin stvaranja računarskog programa naziva se **programiranje**.



HARDVER I SOFTVER

- Da bi se određeni zadatak obavio na računaru potrebni su računarski sistem i odgovarajući programi.
- Sve ono što je u vezi sa fizičkom realizacijom računara naziva se **mašinska oprema** ili **hardver** (*hardware*).
- Sve ono što je u vezi sa programima nazivamo **programska oprema** ili **softver** (*software*).



KOMUNIKACIJA ČOVEK-RAČUNAR

- Na koji način čovek i računar komuniciraju kada znamo da računari razumeju samo jezik binarnih jedinica i nula ili kako se još naziva **mašinski jezik** i reaguju samo na binarne električne signale?



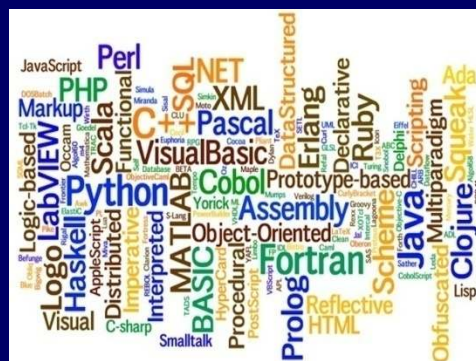
KOMUNIKACIJA ČOVEK-RAČUNAR

- Čoveku bi bilo najprikladnije kada bi sa računarom komunicirao na način koji je najbliži međusobnom komuniciranju ljudi, koristeći prirodne jezike (engleski, srpski itd.).



PROGRAMSKI JEZICI

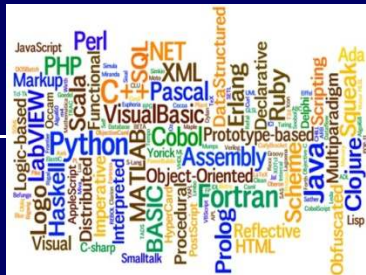
- Da bi se premostile ove razlike u komuniciranju između ljudi i računara - stvoreni su veštački jezici koji se nazivaju **programski jezici**, kao što su FORTRAN, COBOL, PASKAL, C, DELPHI, PHP, VB, JAVA itd.



PROGRAMSKI JEZICI

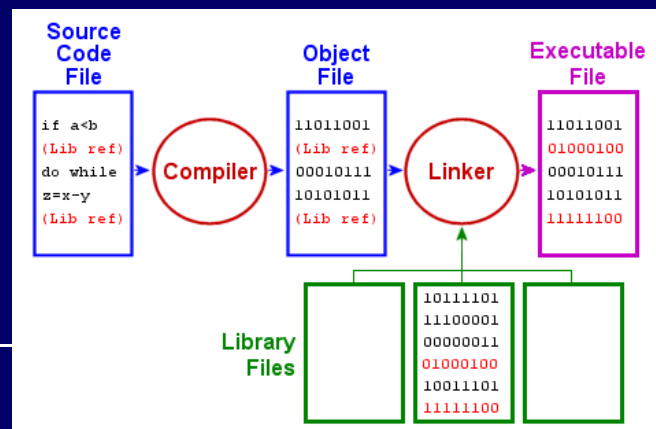
Programski jezici su čoveku mnogo prikladniji od mašinskog jezika, a računaru su bliži od prirodnih jezika, jer nemaju neke nedostatke koje imaju prirodni jezici kao što su:

- zamršena sintaksa,
- komplikovana gramatička pravila,
- česte nejasnoće i dvosmislenosti kod određivanja značenja rečenice (semantike).



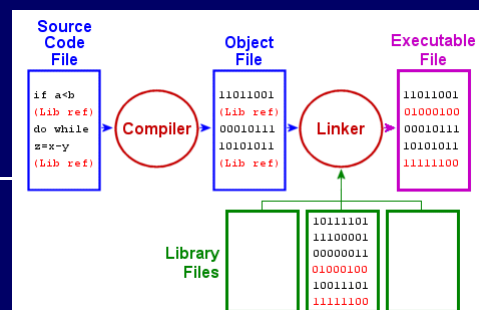
PREVODILAC

- Kako računar razume samo mašinski jezik, neophodno je da se programi napisani u programskom jeziku višeg nivoa prevedu na mašinski jezik.
- Tu funkciju obavlja poseban program za prevođenje koji se naziva **prevodilac**.



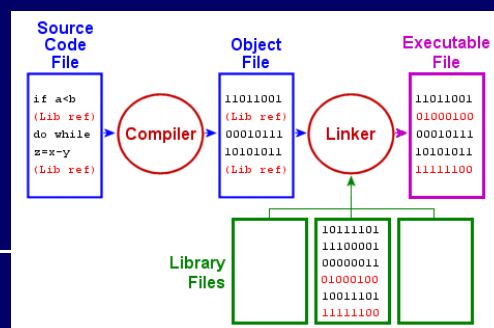
PREVODILAC

- Prevodilac prevodi program sa višeg programskog jezika na mašinski jezik u skladu sa gramatičkim pravilima tog jezika.
- Ukoliko u toku prevođenja prevodilac otkrije gramatičke greške, kaže se da program ima sintaksičke greške.
- Takve geške moraju da se isprave a zatim se program ponovo prevodi.



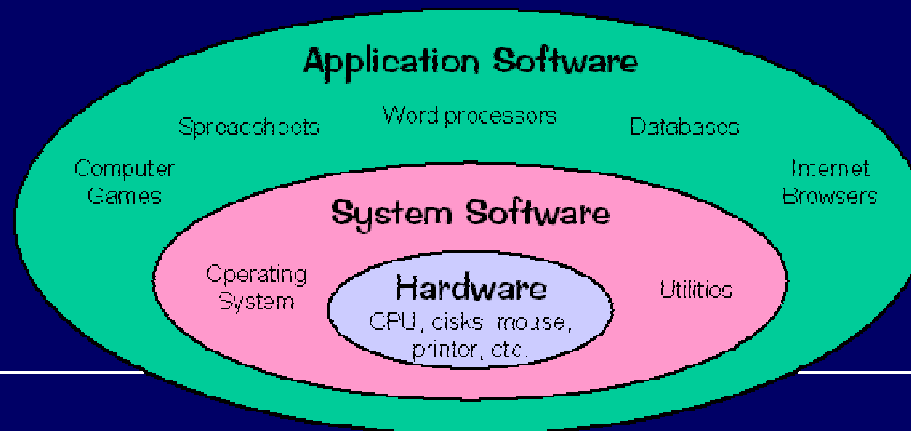
PREVODILAC

- Međutim, postoje i greške koje prevodilac ne može da otkrije a koje za posledicu imaju da izvršavanje programa sa određenim podacima ne daje ispravne rezultate.
- Za ovakve greške kažemo da su semantičke i one se teže otklanjaju.



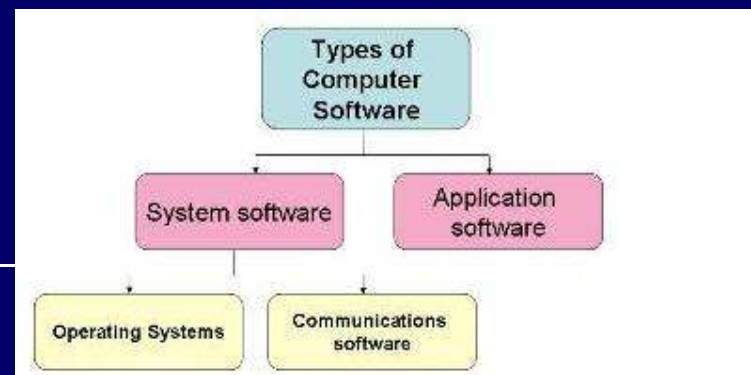
SISTEMSKA PROGRAMSKA PODRŠKA

- Osnovna karakteristika računarskih sistema je integralna povezanost mašinske i programske podrške, odnosno hardvera i softvera.
- Zato je razvoj hardvera pratio i razvoj softvera.



SISTEMSKA PROGRAMSKA PODRŠKA

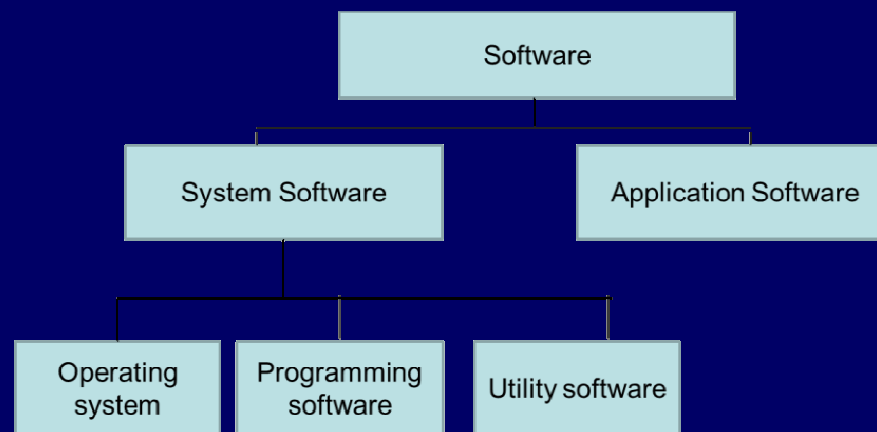
Razvoj sistemske programske podrške kretao se od minimalnog skupa programa, koji je omogućavao samo funkcionisanje računarskog sistema, pa do savremenog sistemskog softvera sa velikim brojem programa koji upravljaju i kontrolišu rad računarskog sistema.





U SISTEMSKU PROGRAMSKU PODRŠKU SPADAJU:

- operativni sistemi
- programi za prevođenje
- uslužni programi.



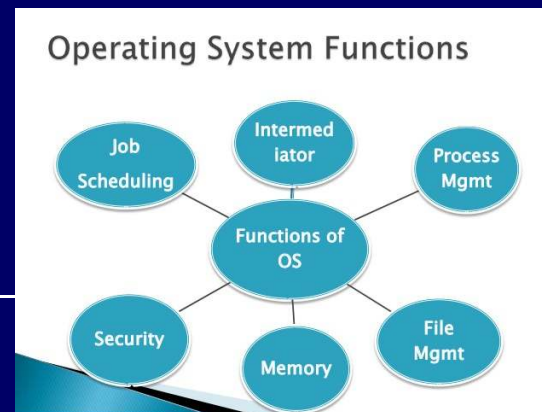
OPERATIVNI SISTEM

- **Operativni sistem** je skup programa koji upravlja radom računara.
- Osnovna funkcija operativnog sistema je da omogući optimalno korišćenje računarskog sistema.



FUNKCIJE OPERATIVNOG SISTEMA

- raspodela hardvera, korisničkih programa i podataka između korisnika računarskog sistema
- kontrola odvijanja ulaznih i izlaznih operacija
- otkrivanje programskih grešaka



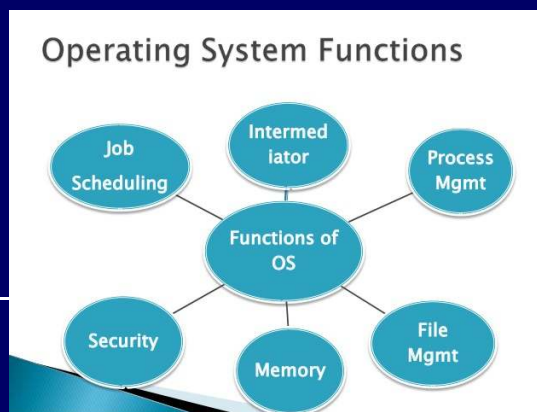
OPERATIVNI SISTEM

- **Operativni sistem** je skup programa koji upravlja radom računara.
- Osnovna funkcija operativnog sistema je da omogući optimalno korišćenje računarskog sistema.



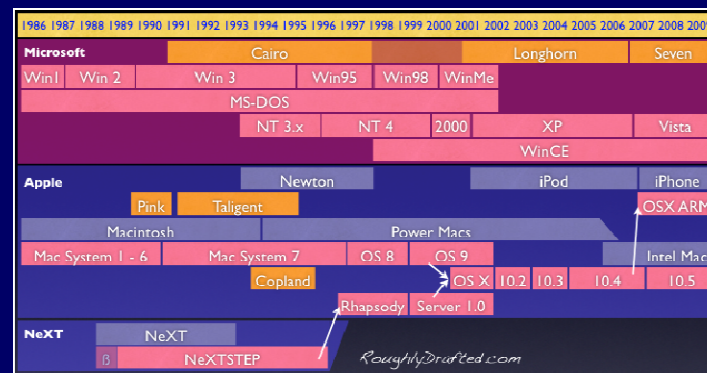
FUNKCIJE OPERATIVNOG SISTEMA

- testiranje hardvera i njegovo povezivanje
- planiranje redosleda odvijanja poslova
- raspodelu vremena centralnog procesora između različitih procesa kod operativnih sistema sa višestrukom obradom itd.



RAZVOJ OPERATIVNIH SISTEMA

- Prvo su nastali operativni sistemi koji su podržavali takozvanu serijsku obradu, gde su se programi izvršavali serijski - jedan za drugim, a operativni sistem je kontrolisao njihovo izvršavanje.



RAZVOJ OPERATIVNIH SISTEMA

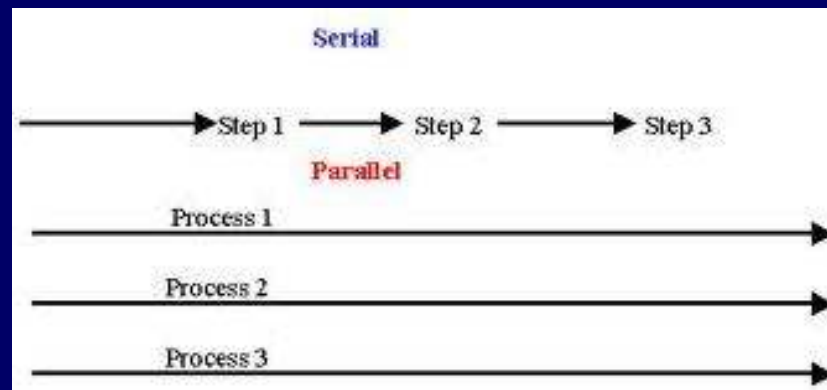
- Savremeni operativni sistemi (Windows, UNIX,...) su veoma kompleksni, ali su mnogo pristupačniji korisnicima.





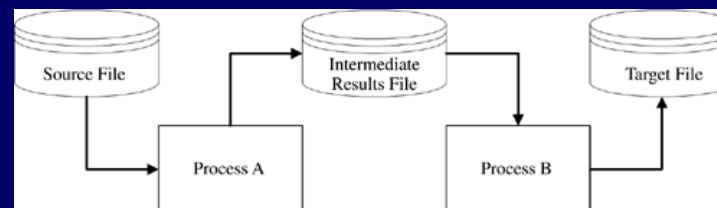
SVI OPERATIVNI SISTEMI MOGU SE PODELITI NA:

- serijske (*batch*) i
- multiprogramske.



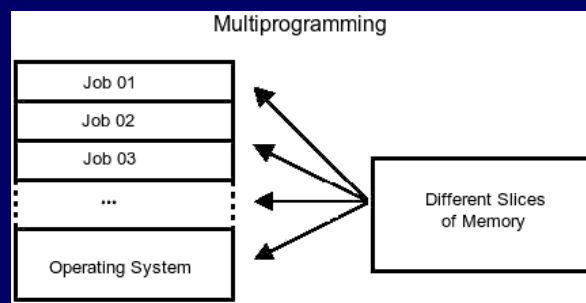
SERIJSKI OPERATIVNI SISTEM

- **Serijski operativni sistem** je jednostavan operativni sistem kod kojeg se programi izvršavaju jedan za drugim, u datom redosledu.
- Dok se ne završi izvršavanje jednog programa ne može da počne izvršavanje drugog programa.



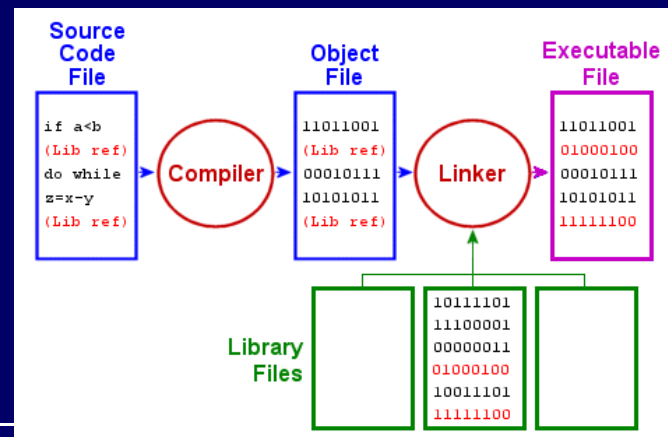
MULTIPROGRAMSKI OPERATIVNI SISTEMI

- **Multiprogramski operativni sistemi** omogućavaju da se više programa nađu istovremeno u glavnoj memoriji i da dele raspoložive resurse.



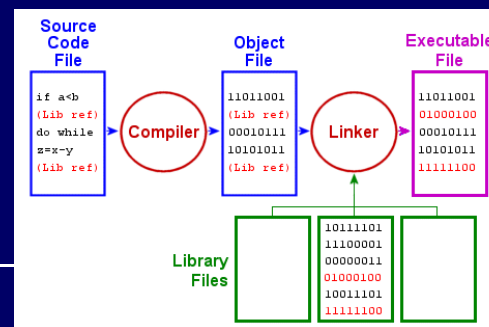
PROGRAM ZA PREVOĐENJE - PREVODILAC (*COMPILER*)

- Program za prevođenje - prevodilac (*compiler*) ima osnovni zadatak da program napisan u nekom višem programskom jeziku prevede na mašinski jezik.



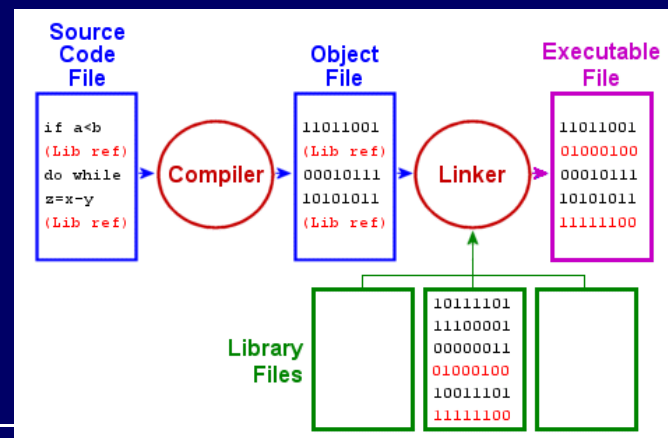
PROGRAM ZA PREVOĐENJE - PREVODILAC (COMPILER)

Međutim rezultat rada prevodioca obično još uvek nije u obliku programa koji se može izvršiti, jer se u višim programskim jezicima upotrebljavaju potprogrami, to jest, jednom instrukcijom se zamenjuje više instrukcija - na primer za standardne funkcije, ulazne i izlazne operacije itd.



PROGRAM ZA POVEZIVANJE (*LINKER*).

- Potrebno je ovim potprogramima, koji se nalaze u biblioteci sistemskih programa, dopuniti prevedeni program.
- Taj zadatak obavlja uslužni program za povezivanje (*linker*).



PROGRAMA ZA PUNJENJE (*LOADER*)

Dopunjen prevedni program se zatim pomoću uslužnog programa za punjenje (*loader*) smešta u memoriju i tek tada je spreman za izvršavanje.

