Опис асемблера

Студент:Тамара Сретковић ментор: Саша Стојановић

Садржај

[Увод 3](#_Toc461973422)

[Упутство за превођење програма 3](#_Toc461973423)

[Упутство за покретање програма 3](#_Toc461973424)

[Опис решења 4](#_Toc461973425)

[Опис релокација 5](#_Toc461973427)

# Увод

Овај асемблер је настао као као одговор на задатак из предмета Системски Софтвер са Електротехниког факултета Универѕитета у Београду.

У директоријуму src се налази изворни код решења, извршна верзија програма (.exe) се налази у директоријуму bin. У истом директоријуму се налазе и тест фајлови за тестирање програма.

Поставка задатка је она из испитног рока за јун 2019. године.

Програм је написан на језику C++ и за оперативни систем Linux.

Превођење и покретање под овим оперативним системом је описано у даљем тексту.

# Упутство за превођење програма

За превођење програма се користи g++ преводилац, верзија 9.

У слуају да немате наведени компајлер на Linux систему под којим радите, могуће је извршити инсталазију из командне линије укуцавањем следећих команди:

sudo add-apt-repository ppa:ubuntu-toolchain-r/test

sudo apt-get update

sudo apt-get install gcc-9

sudo apt-get install g++-9

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-9 50

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/gcc-9 50

sudo update-alternatives --remove-all gcc

sudo update-alternatives --remove-all g++

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-9 50

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-9 50

sudo update-alternatives --config gcc

sudo update-alternatives --config g++

Инсталација имплементираног решења је релативно једноставна и ради се из командне иније.

Потребно је да се лоцирате у директоријуму у коме се налази иворни код решења. Потом, је неопходно укуцати команду make, која ће креирати извршни фајл (make –f Makefile).

Програм покрећете покретањем новонасталог фајла.

# Упутство за покретање програма

Програм се може покренути само из командне линије.

Ово се постиже на стандардан начин за оперативнеи систем Linux. Ако локација изврсног фајла није додата у путању системских програма, неопходно је позиционирати се у директоријум у коме се налази извршна веерзија програма.

Програм се покреће тако што се командној линији зада име програма (./име\_програма, ако није додат у путању системских програма) и путанју до имена текстуалних датотека које садрже код у асемблеру који се треба превести.

Пример покретања: ./asm input.txt output.txt

# Опис решења

Асемблер који једат у овом решењу је двопролазни и имплементиран је на начин који је дат на предавањима из предмета Системски Софтвер.

У првом пролазу асемблер пролази кроз уазну датотеку и нлази лабеле које су дефинисане за сваку секцију. Када наиђе на нову дефиницију асемблер колики је померај те лабеле у односу на почетак текуће секције асемблерског кода и уписује дати симбол у табелу симбола. За један симбол се у табели симбола памти име симбола, секција у којој се налази, померај од почетка текуће секције, величина секције, као и редни број у табели симбола. Поред наведених података у табели се води и евиденција о досегу сваког симбола. Досег је глобалан, или локалан за дати модул. У првом пролазу се досег свих новодефинисаних симбола поставља на локални.

Поред лабела у табелу симбола се и уписују секције.

На крају првог пролаза асембер има делимино попуњену табелу симбола.

Како би се припремио за други пролаз, асемблер се враћа на почетак фајла и ресетује све бројаче секција.

У другом пролазу асемблер врши превођење асемблерског кода на машински јеѕик дат по спецификацији процесора за који је асемблер и писан, а накази се у документу Prilog.doc.

У случају да асемблер наиђе на директиву .global <simbol>, oн ће променити досег симбола “simbol” на глобални.

Све инструкције се декодирају у машинске речи дужине 4 бајта или 8 бајтова. Инструкције, као и начин на који се користе се налазе у табели инструкција у овом документу.

У свим инструкцијама се могу користити дефинисане лабеле. За лабеле дефинисане ван дате секције или које нису дефинисане у датом фајлу ће се извршити релокација. За сваку секцију се води табела релокација. За лабеле које су дефинисане у текућој секцији, пре или после реферисања ће се одмах срачунати померај на основу података из табеле симбола.

Постоји и .еqu директива, која симболичком имену додељује нумеричку константу.

## Опис релокација

У систему постоје две врсте релокација.

Релативне релокације се извршавају ако за инструкције које захтевају промену програмској бројача (било да је то природа саме инструкције, или се користи регистарско индиректно адресирање помоћу операнда ‘$’)

Апсолутне релокације се извршавају у преосталим случајевима, односно када је померај константан.