

Programos kūrimo ciklas ir jo valdymas

Saulius Gražulis

2011 ruduo

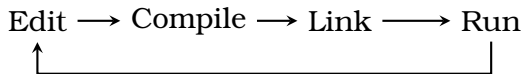
Vilnius University, Faculty of Mathematic and Informatics
Institute of Informatics



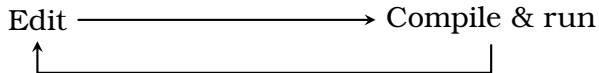
This set of slides may be copied and used as specified in the
[Attribution-ShareAlike 4.0 International](#) license



- Tipiškas *kompilijuojančios* kalbos programos kūrimo (mikro-)ciklas:



- Tipiškas *interpretuojančios* kalbos programos kūrimo (mikro-)ciklas:



Edit-Compile-Link-Run ciklas *x aplinkoje

Kaip visada, viską galima padaryti komandomis

- Paprasta C programa (C++, Ada, Java, Fortran, Pascal, ...)

```
sh> vi main.c
sh> cc main.c -o prog
sh> ./prog arg1 arg2 arg3
```

- Sudėtingesnė C programa

```
sh> vi main.c
sh> cc -c main.c -o main.o
sh> cc -c math.c -o math.o
sh> cc -c libr.c -o libr.o
sh> cc main.o math.o libr.o -o prog -lm
sh> ./prog arg1 arg2 arg3
```

Edit-Compile&Run ciklas *x aplinkoje

Be abejo, viskas vykdoma komandomis

- Perl'o, Python'o arba Shell skriptas (C :-), FORTH, Lisp, Haskel, Prolog, ...)

```
sh> vi script
```

```
sh> ./script arg1 arg2 arg3
```

- build.sh:

```
cc -c main.c -o main.o
cc -c math.c -o math.o
cc -c libr.c -o libr.o
cc main.o math.o libr.o -o prog -lm
./prog arg1 arg2 arg3
```

- GNUmakefile, makefile, Makefile:

```
run: prog
———→./prog arg1 arg2 arg3
prog: main.o math.o libr.o
———→cc main.o math.o libr.o -o prog -lm
main.o: main.c
———→cc -c main.c -o main.o
# ... ir dar komandos
# math.o bei libr.o sukurti.
```

- build.sh:

```
set -e
cc -c *.c
cc *.o -o prog -lm
./prog arg1 arg2 arg3
```

- GNUmakefile, makefile, Makefile:

```
run: prog
----->./prog arg1 arg2 arg3
prog: main.o math.o libr.o
----->cc main.o math.o libr.o -o prog -lm
%.o: %.c
----->cc -c $< -o $@
```

- GNUmakefile, makefile, Makefile:

```
run: prog
-----> ./prog arg1 arg2 arg3
prog: main.o math.o libr.o
-----> cc $^ -o $@ -lm
%.o: %.c
-----> cc -c $< -o $@
```

- GNUmakefile, makefile, Makefile:

```
PROGRAM = prog
run: ${PROGRAM}
    ./${PROGRAM} arg1 arg2 arg3
${PROGRAM}: main.o math.o libr.o
    cc $^ -o $@
%.o: %.c
    cc -c $< -o $@
```


- GNUmakefile, makefile, Makefile:

```
PROGRAM = prog
SOURCES = $(wildcard *.c)
OBJECTS = ${SOURCES:%.c=%.o}
run: ${PROGRAM}
    ./${PROGRAM} arg1 arg2 arg3
${PROGRAM}: ${OBJECTS}
    cc $^ -o $@
%.o: %.c
    cc -c $< -o $@
```

make – pridedame „gerai žinomas tikslus“ (“well known targets”)

- GNUmakefile, makefile, Makefile:

```
PROGRAM = prog
SOURCES = $(wildcard *.c)
OBJECTS = ${SOURCES:%.c=%.o}
all: run
run: ${PROGRAM}
    ./${PROGRAM} arg1 arg2 arg3
${PROGRAM}: ${OBJECTS}
    cc $^ -o $@
%.o: %.c
    cc -c $< -o $@
clean:
    rm -f ${OBJECTS}
cleanAll distclean: clean
    rm -f ${PROGRAM}
```

- GNUmakefile, makefile, Makefile:

```
PROGRAM = prog
SOURCES = $(wildcard *.c)
OBJECTS = ${SOURCES:%.c=%.o}
.PHONY: all run clean cleanAll distclean
all: run
run: ${PROGRAM}
    ./${PROGRAM} ${ARGS}
${PROGRAM}: ${OBJECTS}
    ${CC} ${CFLAGS} $^ -o $@
%.o: %.c
    ${CC} ${CFLAGS} -c $< -o $@
clean:
    rm -f ${OBJECTS}
cleanAll distclean: clean
    rm -f ${PROGRAM}
```

make – universalus Makefile'as

- GNUmakefile, makefile, Makefile:

```
PROGRAM = $(notdir $(shell pwd))
SOURCES = $(wildcard *.c)
OBJECTS = ${SOURCES:%.c=%.o}
.PHONY: all run clean cleanAll distclean
all: run
run: ${PROGRAM}
    ./${PROGRAM} ${ARGS}
${PROGRAM}: ${OBJECTS}
    ${CC} ${CFLAGS} $^ -o $@
%.o: %.c
    ${CC} ${CFLAGS} -c $< -o $@
clean:
    rm -f ${OBJECTS}
cleanAll distclean: clean
    rm -f ${PROGRAM}
```

- sh> cd ~/projektas/prog
sh> vi main.c
sh> vi math.c
sh> make
sh> make CFLAGS=-g
sh> make ARGS="vienas du trys"
sh> make clean
sh> make distclean

Naudingi make parametrai

- `sh> make -n`
`sh> make -p`
`sh> make -W libr.c`

- GNU Make – user's manual

Richard Stallman, Roland McGrath, Paul D. Smith

<http://www.gnu.org/software/make/>

2009-10-20

- Command line

Daug autorių, išvardinti tinklalapiuose:

http://en.flossmanuals.net/command-line/ch055_credits/

2011-10-17

<http://en.flossmanuals.net/command-line/>

2011-10-17

http://en.flossmanuals.net/_booki/command-line/command-line.pdf

2011-10-17