vs code

самому для себя разобраться как

1. запускать обычный main с помощью команды vscode терминала

способ с лекции сударенко через exe

gcc -c -o main.o main.c

gcc -o prog main.o

2. запускать обычный мэйн с помощью makefile

3. запускать составную программу с помощью vscode терминала

4. запускать составную программу с помощью makefile

вебинар 11

makefile:

all: prog

prog: main.o temp\_function.o

gcc -o prog main.o temp\_function.o

main.o: main.c

gcc -c -o main.o main.c

temp\_function.o: temp\_function.c

gcc -c -o temp\_function.o temp\_function.c

clean:

del \*.o

del prog.exe

mingw32-make prog

mingw32-make clean

.\prog.exe -h

.\prog.exe -f temperature\_small.csv -m 2

-f, c:/Users/gladyshevaa/Documents/\_auto/mfti/course\_for\_vs\_code/getopt\_test\_func/temperature\_small.csv, -m, 2

***Cygwin*** — это UNIX-подобная среда и интерфейс командной строки для Microsoft Windows. Программа Cygwin расширяет возможности стандартной командной строки Windows, позволяя автоматизировать некоторые процессы и разрабатывать программы для Linux и других систем.

Список пакетов, которые можно скачать и установить

gcc-core (в разделе Devel)

***gcc-g++ (в разделе Devel)***

***make (в разделе Devel)***

***cmake (в разделе Devel)***

automake (в разделе Devel)

wget (в разделе Web)

libiconv (в разделах Devel и Libs)

openssh (в разделе Net)

nano (в разделе Editors)

bash (из раздела Base)

base-cygwin (из раздела Base)

base-files (из раздела Base)

cygwin (из раздела Base)

file (из раздела Base)

gzip (из раздела Base)

login (из раздела Base)

mintty (из раздела Base)

openssl (из раздела Base)

run (из раздела Base)

tar (из раздела Base)

vim-minimal (из раздела Base)

cron (из раздела Admin)

shutdown (из раздела Admin)

git (все пакеты из раздела Devel)

// При установке он просит выбрать сервер откуда будем скачивать непосредственно пакеты. на выбор серверы из-за рубежа, видно что не официальные источники пакетов.

***CMake*** кроссплатформенное программное средство автоматизации сборки программного обеспечения из исходного кода. ***Не занимается непосредственно сборкой, а лишь генерирует файлы сборки из предварительно написанного файла сценария CMakeLists.txt и предоставляет простой единый интерфейс управления.*** Помимо этого, способно автоматизировать процесс установки и сборки пакетов. Сборка программы или библиотеки с помощью CMake представляет собой двухэтапный процесс. На первом этапе стандартные файлы сборки генерируются из файлов конфигурации (CMakeLists.txt), которые написаны на языке CMake. Затем ***задействуются системные инструменты сборки, такие как make***, Ninja, используемые для непосредственной компиляции программ

Q.: I am a computer engineering student focusing on embedded systems. I’m configuring a new computer for C/C++ development and want to know which environment I should use for the compiler. I’m between Cygwin and MSYS 2. What do you recommend I use for this purpose and why?

A.: I use MINGW from the Windows command line because it doesn't add a cygwin.dll nor msys2.dll dependency. I can copy then run my exe on non-dev machines with no additional configuration.

about gcc

If you have issues where you can't stop on a breakpoint or can't show variable info, you probably need to turn off optimization and turn on debug symbols. When using gcc, the flags are -g3 -O0. This way the compiler won't optimize away any if your code.

Отладка c программы в vsc

<https://stackoverflow.com/questions/77646301/whats-the-difference-between-pressing-f5-and-clicking-the-run-button-at-the-to>

**makefile for debugging**

Везде добавил флаг -g и заработало.

состав проекта

- main.c

- temp\_function.h

- temp\_function.c

all: prog

prog: main.o temp\_function.o

gcc -g -o prog main.o temp\_function.o

main.o: main.c

gcc -g -c -o main.o main.c

temp\_function.o: temp\_function.c

gcc -g -c -o temp\_function.o temp\_function.c

clean:

del \*.o

del prog.exe