

1. 01.hello

2024/7/21 Table of Contents

01.hello

目的

構成データ

例題 hello

課題1 ファイル読み込み

課題2 ファイル書き込み

課題3 シグナル

1.1. 目的

組込みアプリケーション開発 01.hello

1.2. 構成データ

1.2.1. /media/sf_ArmadilloX1/hwpwm/work/R06_2024/Apllication_debug/text/practice ディレクトリ

▼ .../share/ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice/ の構成

```
1 | user@1204PC-Z490M:/mnt/v/VirtualBoxWork/share/ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice$ tr
2 | ./
3 | └─ 01.hello/
4 |   └─ article.txt*           <── 課題1 ファイル読書きの対象ファイル
5 |   └─ count.c*              <── 課題1 ファイル読み込み
6 |   └─ hello.c*              <── 例題
7 |   └─ Makefile*             <── /work/linux/nfsroot/debug/04_practice ヘコピー
8 |   └─ rot13.c*              <── 課題2 ファイル書き込み
9 |   └─ sigtest.c*            <── 課題3 シグナル
10 |
11 |
```

1.3. 例題 hello

Hello,World の表示

1.3.1. ソースファイル

hello.c

▼ hello.c

```
1 | #include <stdio.h>
2 |
3 | int main(void)
4 | {
5 |     printf("Hello ArmadilloX1 \n");
6 |     return 0;
7 | }
```

1.3.2. make から動作確認まで

make

▼ make clean -> make -> sudo make install

```
1 atmark@atde8:/media/sf_ArmadilloX1/hwpwm$ cd /media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/prc
2
3 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make cle
4 rm -f hello
5
6 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make
7
8 arm-linux-gnueabihf-gcc -gdwarf-2 -O0 hello.c -o hello
9
10 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ sudo mak
11 [sudo] atmark のパスワード:
12 cp -p hello /work/linux/nfsroot/debug/04_practice
13 cp -p hello /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
14 cp -p hello.c /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
15
16 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$
```

実行

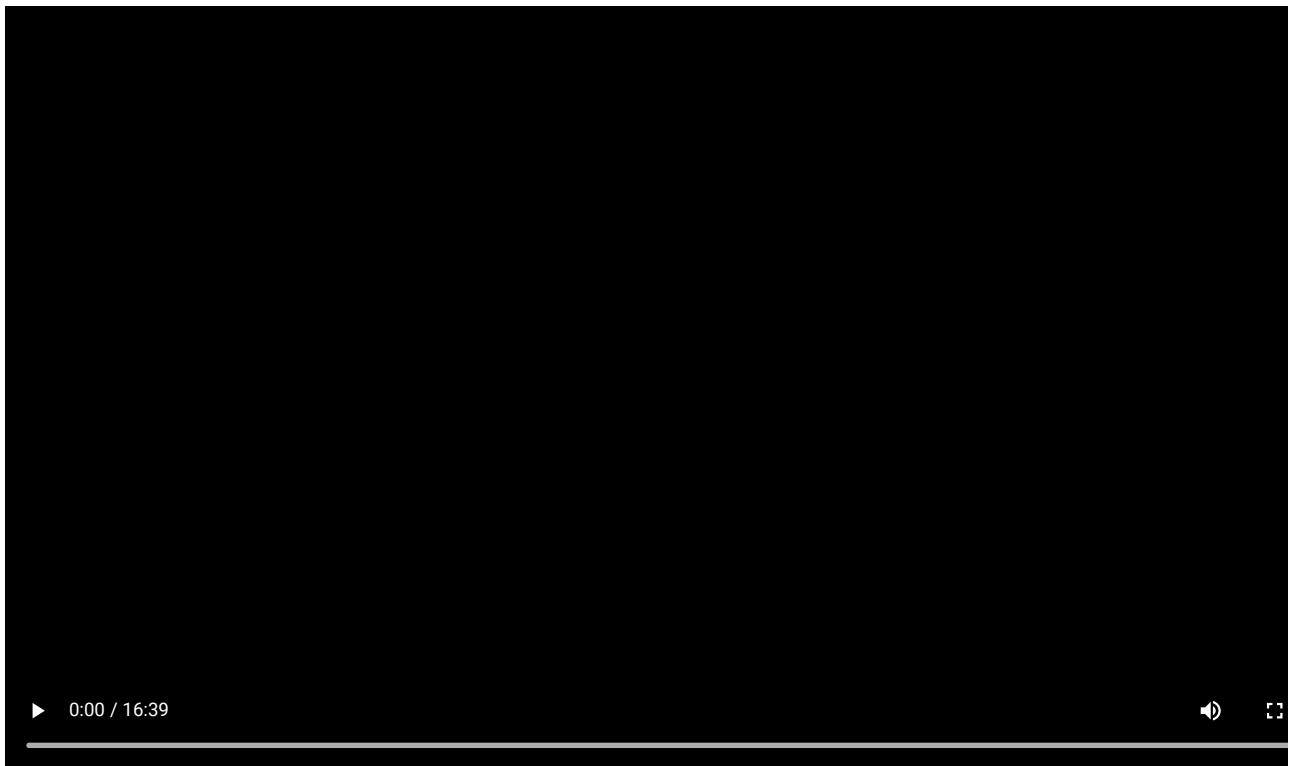
▼ root@armadillo:/debug/04_practice# ./hello

```
1 root@armadillo:/debug/04_practice# ./hello
2 Hello, world!
```

実行している様子

▼ CSIDE と TeraTerm

<https://youtu.be/YVPniqPF-bY>



1.4. 課題1 ファイル読み込み

article.txt ファイルを読み込んで文字数をカウントし表示する

1.4.1. ソースファイル

count.c

▼ count.c

```
1  #include <stdio.h>
2  #include <fcntl.h>
3  #include <string.h>
4  #include <unistd.h>
5
6  #define BUFFER_SIZE    1024
7
8  int main(int argc, char **argv)
9  {
10     int fd;
11     char buffer[BUFFER_SIZE];
12     int n;
13     int count = 0;
14
15     // コマンドライン引数が2つ以外の場合、エラー終了します。
16     if (argc != 2){
17         printf("usage: count file\n");
18         return 2;
19     }
20
21     // ファイルをオープンします。
22     fd = XXXXX(XXXXX, XXXXX);
23     // ファイルをオープンできなかった場合、エラー終了します。
24     if (fd < 0){
25         perror("open");
26         return 1;
27     }
28
29     // ファイルの中身を読み込み、文字数をカウントします。
30     while (n = XXXXX(fd, XXXXX, BUFFER_SIZE - 1)){
31         if (n < 0){
32             perror("read");
33             return 1;
34         }
35         buffer[n] = '\0';
36         count += strlen(buffer);
37     }
38
39     // 文字数を表示します。
40     printf("%d\n", count);
41
42     // ファイルをクローズします。
43     XXXXX(fd);
44
45     return 0;
46 }
47
```

article.txt

▼ root@armadillo:/debug/04_practice# cat article.txt

```

1 root@armadillo:/debug/04_practice# cat article.txt
2 Big-Game Hunt Adds to Evidence of Early North American Settlement
3
4 For many years, it was thought that the Clovis people were the first humans to populate North America, about 13,000
5
6 But recently, evidence has suggested that other settlers arrived earlier, and a new study lends support to that theory.
7
8 The study, in the journal Science, finds that a mastodon rib with a bone point lodged in it dates back 13,800 years.
9
10 "It's the first hunting weapon found pre-Clovis," said the lead author, Michael R. Waters, an archaeologist at Texas A&M University.
11
12 The fossils had been discovered in the late 1970s near Manis, Wash., by Carl Gustafson, an archaeologist at Washington State University.
13
14 His theory was questioned by other scientists. But carbon dating technology has improved since then, and Dr. Waters' theory has gained traction.
15
16 They also used CT scanning to closely study the embedded bone point to confirm that it was a hunting tool. They found that it was a
17
18 "It couldn't have been anything else," Dr. Waters said.
19
20 Like the Clovis people, the Manis inhabitants also probably migrated to North America from northeastern and Central Asia.
21
22 "What's nice about all the pre-Clovis sites is that some had stone tools, and now here we see bone tools, all the way back to the
23
24 The Clovis used these basic technologies to create a more sophisticated set of tools, he added.

```

1.4.2. make から動作確認まで

 article.txt を /work/linux/nfsroot/debug/04_practice へコピーすること

make

▼ make clean -> make -> sudo make install

```

1 atmark@atde8:/media/sf_ArmadilloX1/hwpwm$ cd /media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello
2
3 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make clean
4 rm -f hello
5
6 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make
7
8 arm-linux-gnueabi-gcc -gdwarf-2 -O0 hello.c -o hello
9
10 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ sudo make install
11 [sudo] atmark のパスワード:
12 cp -p hello /work/linux/nfsroot/debug/04_practice
13 cp -p hello /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
14 cp -p hello.c /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
15
16 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$

```

article.txt を /work/linux/nfsroot/debug/04_practice へコピー

▼ atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello\$ sudo cp article.txt /work/linux/nfsroot/debug/04_practice/

```

1 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ sudo cp article.txt /work/linux/nfsroot/debug/04_practice/

```

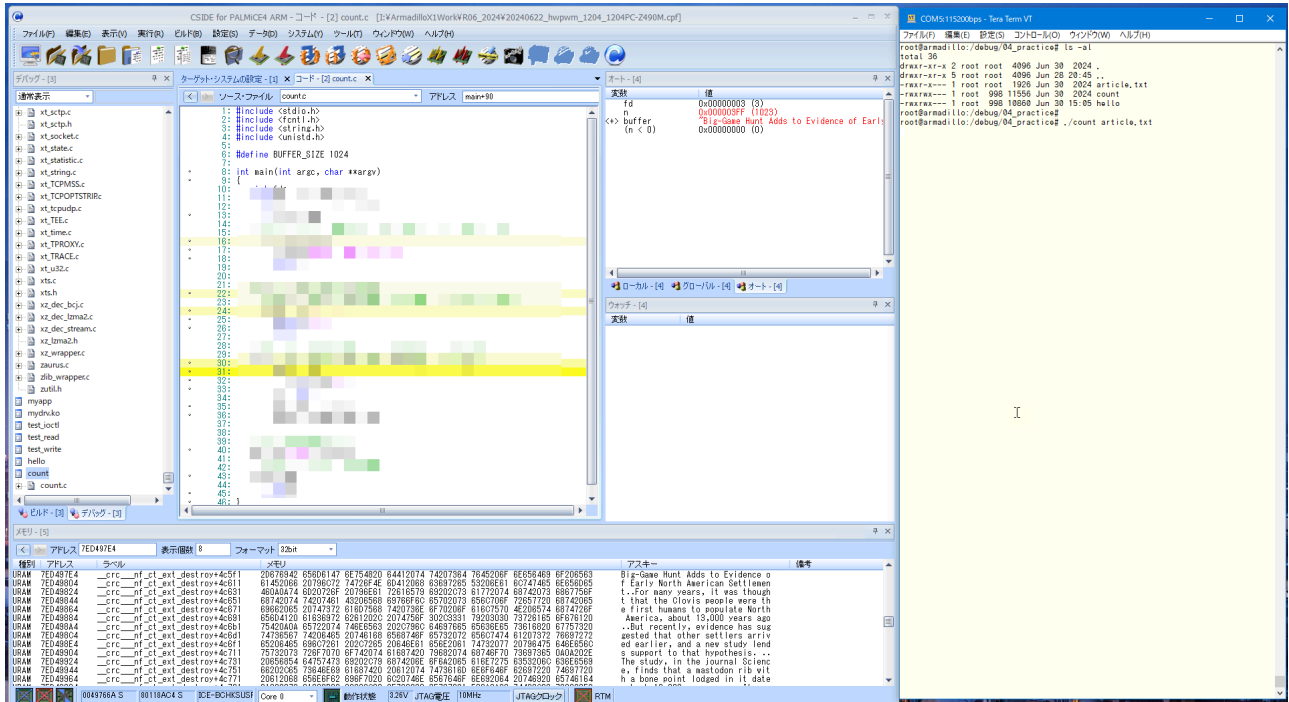
実行

▼ root@armadillo:/debug/04_practice# ./count article.txt

```
1 | root@armadillo:/debug/04_practice# ./count article.txt
2 | 1926 <----- article.txt ファイルの文字数
```

実行している様子

▼ CSIDE と TeraTerm



1.4.3. ヒント

open, read, close

1.5. 課題2 ファイル書込み

- ROT13暗号化方式によってアルファベットを13文字進める
- AはN,BはO, ... ,ZはM,aはn,bはo, ... ,zはm
- アルファベット以外の文字は変更しない
- article.txt ファイルを読込んで、結果は article.rot へ書込む
- 実行方法は ./rot13 article.txt article.rot

1.5.1. ソースファイル

rot13.c

▼ rot13.c

```
1  #include <stdio.h>
2  #include <fcntl.h>
3  #include <unistd.h>
4
5  #define BUFFER_SIZE    1024
6
7  int main(int argc, char **argv)
8  {
9
10
11
12
13
14
15
16
17
18
19
20
21      // 第2引数で指定されたファイルをオープンします。
22
23
24
25
26
27
28
29      // 第3引数で指定されたファイルをオープンします。
30
31
32
33
34
35
36
37      // 第2引数で指定されたファイルの中身を読み込んで、読み込んだ文字を13文字進めます。
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62      // ファイルをクローズします。
63
64
65
```

```

65
66         return 0;
67     }
68
69

```

1.5.2. make から動作確認まで

make

▼ make clean -> make -> sudo make install

```

1  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make cle
2  rm -f count
3
4  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make
5  arm-linux-gnueabi-gcc -gdwarf-2 -O0    rot13.c    -o rot13
6
7  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ sudo mak
8  [sudo] atmark のパスワード:
9  cp -p rot13 /work/linux/nfsroot/debug/04_practice
10 cp -p rot13 /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
11 cp -p rot13.c /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
12
13 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$

```

実行

▼ root@armadillo:/debug/04_practice# ./rot13 article.txt article.rot

```

1  | root@armadillo:/debug/04_practice# ./rot13 article.txt article.rot

```

▼ root@armadillo:/debug/04_practice# cat article.rot

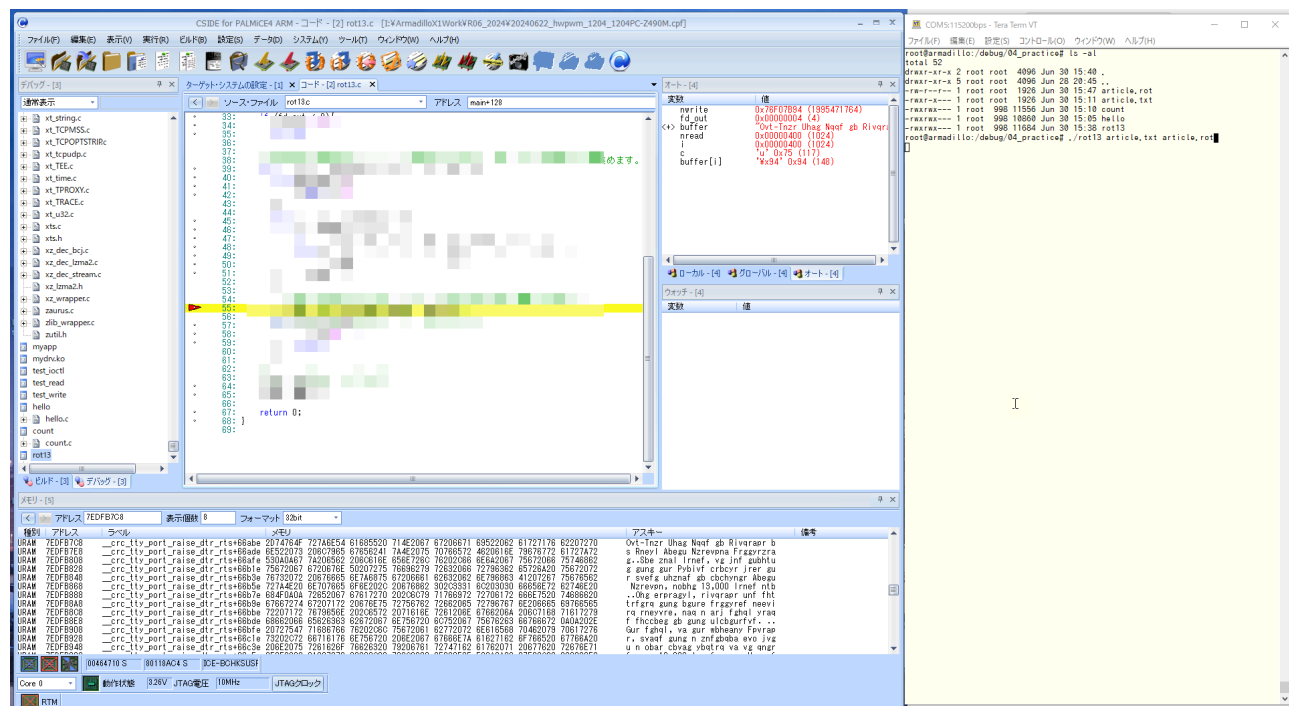
```

1  root@armadillo:/debug/04_practice# cat article.rot
2  Ovt-Tnzt Uhag Nqaf gb Rivqrpr bs Rneyl Abegu Nzrevpna Frggryzrag
3
4  Sbe znal lrnef, vg jnf gubhtug gung gur Pybivf crbcyr jrer gur svefg uhznaf gb cbchyngv Abegu Nzrevpn, nobhg 13,0
5
6  Ohg erpragyl, rivqrpr unf fhtrfrgr gung bgure frggryef neevirg rneyvre, naq n arj fghql yraqf fhccbeg gb gung u
7
8  Gur fghql, va gur wbheany Fpvrpr, svaqf gung n znfgbqba evo jvgu n obar cbvag ybqtrq va vg qngrf onpx 13,800 lrr
9
10 "Vg'f gur svefg uhagvat jrncba sbhaq cer-Pybivf," fnvq gur yrnq nhgube, Zvpunry E. Jngref, na nepunrbybtvfg ng Gr
11
12 Gur sbffvyf unq orra qvfpbirerq va gur yngr 1970f arne Znavf, Jnfu., ol Pney Thfgnsfba, na nepunrbybtvfg ng Jnfuv
13
14 Uvf gurbel jnf dhrfgvbarq ol bgure fpvragvfgf. Ohg pneoba qngvat grpuabybtl unf vzcebirg fvapr gura, naq Qe. Jngr
15
16 Gurl nyfb hfrq PG fpnaavat gb pybfryl fghql gur rzorqqrq obar cbvag gb pbasvez gung vg jnf n uhagvat gbby. Gurl s
17
18 "Vg pbhyqa'g unir orra nalguvat ryfr," Qe. Jngref fnvq.
19
20 Yvxr gur Pybivf crbcyr, gur Znavf vaunovgnagf nyfb cebonoyl zvtengrq gb Abegu Nzrevpn sebz abegurnfgrea naq Prage
21
22 "Jung'f avpr nobhg nyy gur cer-Pybivf fvgrf vf gung fbzr unq fgbar gbbyf, naq abj urer jr frr obar gbbyf, nyy gur
23
24 Gur Pybivf hfrq gurfr onfvp grpuabybtrf gb perngr n zber fbcuvfgvpngrq frg bs gbbyf, ur nqqrq.
25 root@armadillo:/debug/04_practice#

```


実行している様子

▼ CSIDE と TeraTerm



1.5.3. ヒント

write

1.6. 課題3 シグナル

- 実行すると"Hello, world!"と表示して一時停止
- Ctrl + c を入力すると"Goodbye, world!"と表示してプログラムを終了

1.6.1. ソースファイル

sigtest.c

▼

```

1  #include <stdio.h>
2  #include <signal.h>
3  #include <unistd.h>
4
5  // シグナルハンドラの処理(ctrl+cが押された時の処理)。
6  void sig_handler(int sig)
7  {
8
9  }
10
11 int main(void)
12 {
13
14
15
16
17
18     // シグナルハンドラを登録します。
19
20
21
22
23
24
25     // シグナルを待ちます。
26     pause();
27
28     return 0;
29 }

```

1.6.2. make から動作確認まで

make

▼ make clean → make → sudo make install

```

1  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make cle
2  rm -f sigtest
3
4  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ make
5  arm-linux-gnueabi-gcc -gdwarf-2 -O0    sigtest.c  -o sigtest
6
7  atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$ sudo mak
8  [sudo] atmark のパスワード:
9  cp -p sigtest /work/linux/nfsroot/debug/04_practice
10 cp -p sigtest /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
11 cp -p sigtest.c /media/sf_ArmadilloX1/hwpwm/dbg/debug_share_hwpwm/R06_2024/04_practice
12
13 atmark@atde8:/media/sf_ArmadilloX1/hwpwm/work/R06_2024/Application_debug/text/practice-example/01.hello$

```

実行

▼ root@armadillo:/debug/04_practice# ./sigtest

```

1  root@armadillo:/debug/04_practice# ./sigtest
2  Hello, world!
3  ^C Goodbye, world! <—— Ctrl + c
4  root@armadillo:/debug/04_practice#

```

実行している様子

▼ CSIDE と TeraTerm

<https://youtu.be/VTt82rHYxgQ>



1.6.3. ヒント

sigaction、 pause