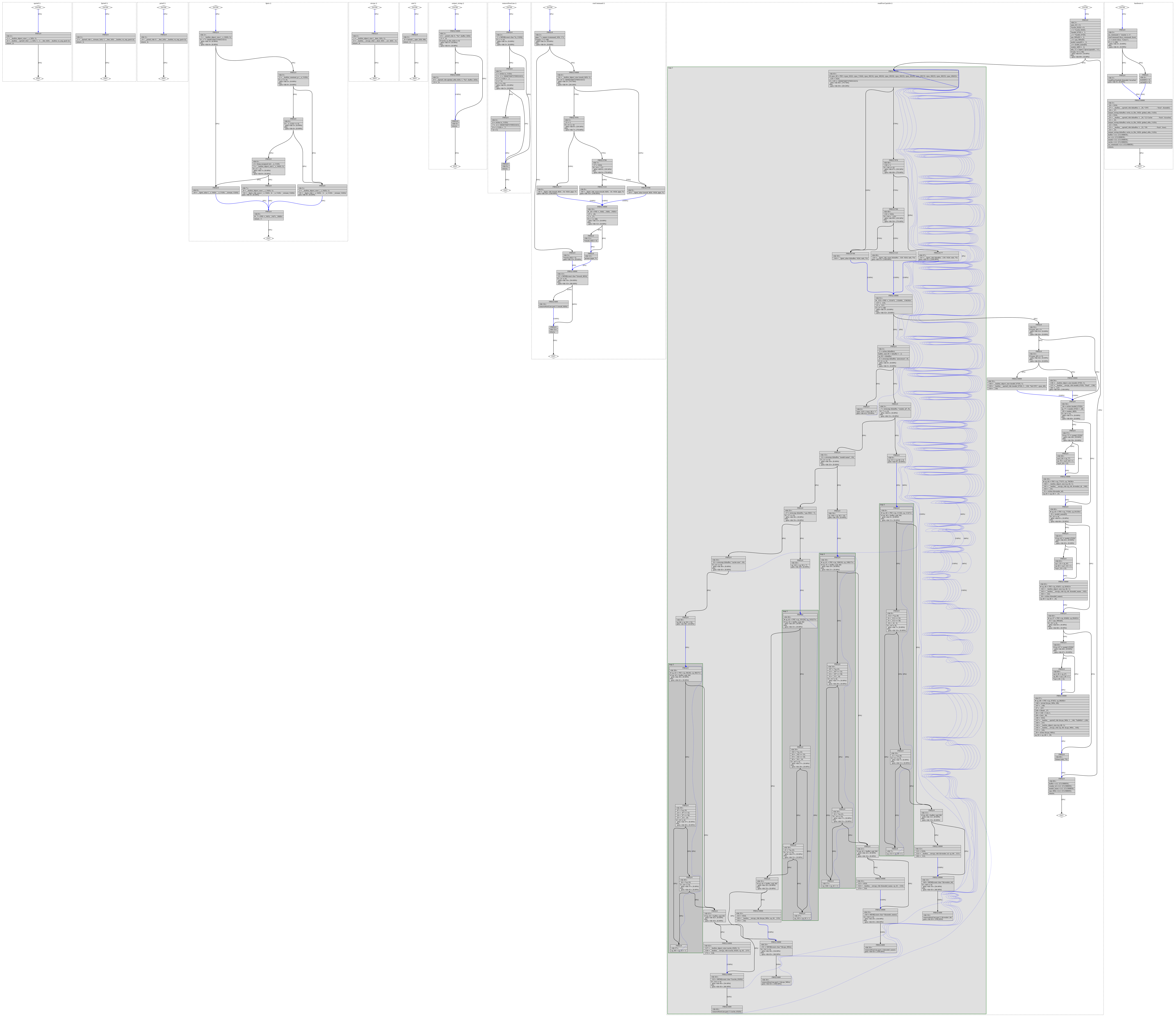

sprintf ()

```
FREQ:0
<bb 2>:
_1 = __builtin_object_size (__s_3(D), 1);
_6 = __builtin___sprintf_chk (__s_3(D), 1, _1, __fmt_4(D), __builtin_va_arg_pack ());
return _6;
```

fprintf ()

```
FREQ:0
<bb 2>:
_5 = __fprintf_chk (__stream_2(D), 1, __fmt_3(D), __builtin_va_arg_pack ());
return _5;
```

printf ()

```
FREQ:0
<bb 2>:
_4 = __printf_chk (1, __fmt_2(D), __builtin_va_arg_pack ());
return _4;
```

fgets ()

```
FREQ:0
<bb 2>:
_1 = __builtin_object_size (__s_10(D), 1);
if (_1 != 18446744073709551615)
  goto <bb 3>; [0.00%]
else
  goto <bb 8>; [0.00%]

FREQ:0
<bb 3>:
_2 = __builtin_constant_p (__n_11(D));
if (_2 == 0)
  goto <bb 5>; [0.00%]
else
  goto <bb 4>; [0.00%]

FREQ:0
<bb 4>:
if (__n_11(D) <= 0)
  goto <bb 5>; [0.00%]
else
  goto <bb 6>; [0.00%]

FREQ:0
<bb 6>:
_4 = (long unsigned int) __n_11(D);
_5 = __builtin_object_size (__s_10(D), 1);
if (_4 > _5)
  goto <bb 7>; [0.00%]
else
  goto <bb 8>; [0.00%]

FREQ:0
<bb 8>:
_18 = __fgets_alias (__s_10(D), __n_11(D), __stream_12(D));

FREQ:0
<bb 7>:
_6 = __builtin_object_size (__s_10(D), 1);
_14 = __fgets_chk_warn (__s_10(D), _6, __n_11(D), __stream_12(D));
goto <bb 9>; [0.00%]

FREQ:0
<bb 5>:
_3 = __builtin_object_size (__s_10(D), 1);
_16 = __fgets_chk (__s_10(D), _3, __n_11(D), __stream_12(D));
goto <bb 9>; [0.00%]

FREQ:0
<bb 9>:
# _7 = PHI <_16(5), _14(7), _18(8)>
return _7;
```

strcpy ()

```
FREQ:0
<bb 2>:
_1 = __builtin_object_size (__dest_3(D), 1);
_6 = __builtin___strcpy_chk (__dest_3(D), __src_4(D), _1);
return _6;
```

atof ()

```
FREQ:0
<bb 2>:
_4 = strtod (__nptr_2(D), 0B);
return _4;
```

output_string ()

```
FREQ:10000
<bb 2>:
_8 = __printf_chk (1, "%s", buffer_3(D));
_11 = _8;
if (write_to_file_5(D) != 0)
  goto <bb 3>; [0.00%]
else
  goto <bb 4>; [0.00%]

FREQ:10000
<bb 3>:
_9 = __fprintf_chk (global_ofile_6(D), 1, "%s", buffer_3(D));
_13 = _9;

FREQ:0
<bb 4>:
return;
```

removeNewLine ()

```
FREQ:0
<bb 2>:
_1 = MEM[(const char *)s_11(D)];
if (_1 != 0)
  goto <bb 3>; [0.00%]
else
  goto <bb 5>; [0.00%]

FREQ:0
<bb 3>:
_2 = strlen (s_11(D));
_3 = _2 + 18446744073709551615;
_4 = s_11(D) + _3;
_5 = *_4;
if (_5 == 10)
  goto <bb 4>; [0.00%]
else
  goto <bb 5>; [0.00%]

FREQ:0
<bb 4>:
_6 = strlen (s_11(D));
_7 = _6 + 18446744073709551615;
_8 = s_11(D) + _7;
*_8 = 0;

FREQ:0
<bb 5>:
return;
```

runCommand ()

```
FREQ:0
<bb 2>:
pipe_7 = popen (command_5(D), "r");
if (pipe_7 == 0B)
  goto <bb 3>; [0.00%]
else
  goto <bb 4>; [0.00%]

FREQ:10000
<bb 4>:
_14 = __builtin_object_size (result_8(D), 1);
if (_14 != 18446744073709551615)
  goto <bb 5>; [79.76%]
else
  goto <bb 9>; [20.24%]

FREQ:7976
<bb 5>:
_15 = 1;
if (_15 == 0)
  goto <bb 6>; [29.56%]
else
  goto <bb 7>; [70.44%]

FREQ:3799
<bb 7>:
_17 = 1024;
if (_14 < _17)
  goto <bb 8>; [29.56%]
else
  goto <bb 9>; [70.44%]

FREQ:4177
<bb 6>:
_16 = __fgets_chk (result_8(D), _14, 1024, pipe_7);
goto <bb 10>; [100.00%]

FREQ:1123
<bb 8>:
_18 = __fgets_chk_warn (result_8(D), _14, 1024, pipe_7);
goto <bb 10>; [100.00%]

FREQ:4700
<bb 9>:
_19 = __fgets_alias (result_8(D), 1024, pipe_7);

FREQ:10000
<bb 10>:
# _20 = PHI <_16(6), _18(8), _19(9)>
_27 = _20;
_1 = _27;
if (_1 == 0B)
  goto <bb 11>; [0.00%]
else
  goto <bb 12>; [0.00%]

FREQ:0
<bb 11>:
*result_8(D) = 0;

FREQ:0
<bb 3>:
*result_8(D) = 0;
goto <bb 13>; [0.00%]

FREQ:0
<bb 12>:
pclose (pipe_7);

FREQ:10000
<bb 13>:
_21 = MEM[(const char *)result_8(D)];
if (_21 != 0)
  goto <bb 14>; [54.00%]
else
  goto <bb 15>; [46.00%]

FREQ:5400
<bb 14>:
removeNewLine.part.1 (result_8(D));

FREQ:0
<bb 15>:
return;
```

readProcCpuInfo ()

```
FREQ:0
<bb 2>:
cpus_63 = 0;
*cache_65(D) = 0;
_1 = *cache_65(D);
*model_67(D) = _1;
_2 = *model_67(D);
cpu_MHz[0] = _2;
_3 = cpu_MHz[0];
model_name[0] = _3;
_4 = model_name[0];
vendor_id[0] = _4;
info_73 = fopen ("/proc/cpuinfo", "r");
if (info_73 != 0B)
  goto <bb 45>; [0.00%]
else
  goto <bb 69>; [0.00%]
```

hardware ()

```
FREQ:0
<bb 2>:
os_command = "uname -s -r";
runCommand (&os_command, &os);
_1 = strstr (&os, "Linux");
if (_1 != 0B)
  goto <bb 3>; [0.00%]
else
  goto <bb 4>; [0.00%]

FREQ:0
<bb 3>:
readProcCpuInfo (&model, &cache);
goto <bb 5>; [0.00%]

FREQ:0
<bb 4>:
model[0] = 0;
cache[0] = 0;

FREQ:10000
<bb 5>:
_26 = 1024;
_27 = __builtin___sprintf_chk (&buffer, 1, _26, "CPU                : %s\n", &model);
_29 = _27;
output_string (&buffer, write_to_file_10(D), global_ofile_11(D));
_24 = 1024;
_25 = __builtin___sprintf_chk (&buffer, 1, _24, "L2 Cache           : %s\n", &cache);
_31 = _25;
output_string (&buffer, write_to_file_10(D), global_ofile_11(D));
_22 = 1024;
_23 = __builtin___sprintf_chk (&buffer, 1, _22, "OS                 : %s\n", &os);
_33 = _23;
output_string (&buffer, write_to_file_10(D), global_ofile_11(D));
buffer ={v} {CLOBBER};
os ={v} {CLOBBER};
model ={v} {CLOBBER};
cache ={v} {CLOBBER};
os_command ={v} {CLOBBER};
return;
```