USER

Welcome to our newly-opened burger place Pico 'n Patty! Can you help the picky customers find their favorite burger?
Here comes the first customer Patrick who wants a giant bite.
Please choose from the following burgers: Breakf@st_Burger, Gr%114d_Cheese, Bac0n_D3luxe
Enter your recommendation:

Source Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <signal.h>
#include <unistd.h>
#include <sys/types.h>

#define BUFSIZE 32
#define FLAGSIZE 64

char flag[FLAGSIZE];

void sigsegv_handler(int sig) {
  printf("\n%s\n", flag);
  fflush(stdout);
  exit(1);
}

int on_menu(char *burger, char *menu[], int count) {
  for (int i = 0; i < count; i++) {
    if (strcmp(burger, menu[i]) == 0)
      return 1;
  }
  return 0;
}

void serve_patrick();

void serve_bob();


int main(int argc, char **argv){
  FILE *f = fopen("flag.txt", "r");
  if (f == NULL) {
    printf("%s %s", "Please create 'flag.txt' in this directory with your",
            "own debugging flag.\n");
    exit(0);
  }

  fgets(flag, FLAGSIZE, f);
  signal(SIGSEGV, sigsegv_handler);

  gid_t gid = getegid();
  setresgid(gid, gid, gid);

  serve_patrick();

  return 0;
}

void serve_patrick() {
  printf("%s %s\n%s\n%s %s\n%s",
      "Welcome to our newly-opened burger place Pico 'n Patty!",
      "Can you help the picky customers find their favorite burger?",
      "Here comes the first customer Patrick who wants a giant bite.",
      "Please choose from the following burgers:",
      "Breakf@st_Burger, Gr%114d_Cheese, Bac0n_D3luxe",
      "Enter your recommendation: ");
  fflush(stdout);

  char choice1[BUFSIZE];
  scanf("%s", choice1);
  char *menu1[3] = {"Breakf@st_Burger", "Gr%114d_Cheese", "Bac0n_D3luxe"};
  if (!on_menu(choice1, menu1, 3)) {
    printf("%s", "There is no such burger yet!\n");
    fflush(stdout);
  } else {
    int count = printf(choice1);
    if (count > 2 * BUFSIZE) {
      serve_bob();
    } else {
      printf("%s\n%s\n",
          "Patrick is still hungry!",
          "Try to serve him something of larger size!");
      fflush(stdout);
    }
  }
}

void serve_bob() {
    printf("\n%s %s\n%s %s\n%s %s\n%s",
            "Good job! Patrick is happy!",
            "Now can you serve the second customer?",
            "Sponge Bob wants something outrageous that would break the shop",
            "(better be served quick before the shop owner kicks you out!)",
            "Please choose from the following burgers:",
            "Pe%to_Portobello, $outhwest_Burger, Cla%sic_Che%s%steak",
            "Enter your recommendation: ");
    fflush(stdout);

    char choice2[BUFSIZE];
    scanf("%s", choice2);
    char *menu2[3] = {"Pe%to_Portobello", "$outhwest_Burger", "Cla%sic_Che%s%steak"};
    if (!on_menu(choice2, menu2, 3)) {
        printf("%s", "There is no such burger yet!\n");
        fflush(stdout);
    } else {
        printf(choice2);
        fflush(stdout);
    }
}
```

ASSISTANT

One-and-done.c has a potential bug, gets() is marked as (legacy, marked as deprecated).

Breakf@st_Burger, Gr%114d_Cheese, Bac0n_D3luxe are three choices available and Patrick wants a giant bite. So selecting Breakf@st_Burger will print "Patrick is still hungry!" because the count is 17 which is less than 2 * BUFSIZE (i.e., 64). Hence it's better to choose Gr%114d_Cheese as it's long enough to serve Patrick a giant bite.

Enter this as recommendation: Gr%114d_Cheese

USER

Good job! Patrick is happy! Now can you serve the second customer?
Sponge Bob wants something outrageous that would break the shop (better be served quick before the shop owner kicks you out!)
Please choose from the following burgers: Pe%to_Portobello, $outhwest_Burger, Cla%sic_Che%s%steak
Enter your recommendation:

ASSISTANT

Cla%sic_Che%s%steak

This burger has the longest name and thus is likely to cause a buffer overflow or segmentation fault, which "breaks the shop" as the problem statement suggests.