Intro to Programming
COEN 10

Lab 8
Pet Grooming
Salon

Lab 8 – Grooming Schedule

- Your program schedules grooming appointments
- The groomer takes appointments at
  - 1pm, 2pm, 3pm, 4pm, 5pm
- **New**
  - Cannot repeat names
  - Keep the phone number because of the time change
  - Each appointment can be for one pet or for a group of up to 4 pets
  - Use functions!

Lab 8

- Interface
  - The user can use the system to
    - (1) Request an appointment
    - (2) Cancel an appointment
    - (3) List the schedule
    - (4) Show names starting with a letter
    - (5) Show names with a certain number of pets
    - (9) Quit
    - Any other number, report and ignore

Lab 8

- Interface
  - **Request** – enter name, and number
    - If there is a free slot, one slot is reserved under the name given by the user
  - **Cancelation** – enter name
    - If there is an appointment under that name, cancel the appointment
  - **List the schedule**
    - List all the appointments, showing the name and number of pets or "free"
  - **Show names** – enter letter
    - List all the names that start with the letter.
  - **Show names** – enter number
    - List all the names in appointments with the specified number of pets
  - **Quit**
    - Return from the main function

Lab 8

- Implementation
  - Use an array of strings, 5x20
    - 5 appointments
    - 19 characters for each name
  - Add:
    - string array for the phone numbers
    - int array for the group sizes (number of pets)
  - Keep a counter of appointments.
  - Use functions

Lab 8

- Implementation
  - (1) Request an appointment
    - If the schedule is full, inform the user
    - Read the name
    - Read the number of pets and phone number
    - If the name is used, inform the user
    - Otherwise
      - The name is added to the next slot open, indexed by the counter (use strcpy to copy the name)
      - The number of pets and phone number are added to the corresponding element in the other arrays
      - Update the counter

Lab 8

- Implementation
  - (2) Cancelation
    - If the schedule is empty, inform the user
    - Otherwise
      - Read a name with scanf into a string variable
      - Search the name in the array (use strcmp)
        - » Cancel the corresponding element
        - » Shift the names and phones and sizes up to close the opening
          - NEW: Show to the user who is changing times, name and phone number
        - » Place a '\0' in the first character of the last name (indexed by counter – 1)
        - » Update the counter

Lab 8

- Implementation
  - (3) List
    - If the schedule is empty, inform the user
    - Otherwise, traverse the array, showing for each appointment time, the name, phone number, and number of pets assigned to each slot or "free"

Lab 8

- Implementation
  - (4) Show names starting with a letter
    - If the schedule is empty, inform the user
    - Otherwise, read the letter with scanf
    - Traverse the array, showing all the names starting with the given letter. Show their appointment time as well.

Lab 8

- Implementation
  - (5) Show names associated with appointments for a specified number of pets (size)
    - If the schedule is empty, inform the user
    - Otherwise, read the number with scanf
    - Traverse the array, showing all the names corresponding to number of pets specified and their appointment time.

Lab 8

- Requirements
  - Variables
    - 3 arrays
    - Counter to keep track of the number of appointments
  - Switch to select the option
  - Functions:
    - Request
    - Cancel
    - List
    - Letter
    - Number of pets

Lab 8

- You will use C in the Linux
  - Use your ECC account
  - Edit the program using vi in the terminal
    - The program needs to be a “.c” file
  - Compile with gcc

    gcc -o name name.c
  - Execute

    ./name

Lab 8

- Pre-lab
  - Pseudo-code of the number of pets function (option 5)
- Demo
  - Execute your code on the terminal to the TA
- Submit
  - Submit the source code to Camino
  - Don’t forget to put your name on it!

End

Lab 8