CSc 3320: Systems Programming

Spring 2021

Midterm 2: Total points = 100

Assigned: 11th Apr 2021, Sunday 11:59 PM

Submission Deadline: 18th Apr 2021, Sunday, 11.59 PM

(No extensions. If your submission is not received by this time then it will NOT be accepted.)

Submission instructions:

1. Create a Google doc for your submission.
2. Start your responses from page 2 of the document and copy these instructions on page 1.
3. Fill in your name, campus ID and panther # in the fields provided. If this information is missing TWO POINTS WILL BE DEDUCTED.
4. Keep this page 1 intact. If this *submissions instructions* page is missing in your submission TWO POINTS WILL BE DEDUCTED.
5. Start your responses to each QUESTION on a new page.
6. If you are being asked to write code copy the code into a separate txt file and submit that as well. The code should be executable. E.g. if asked for a C script then provide myfile.c so that we can execute that script. In your answer to the specific question, provide the steps on how to execute your file (like a ReadMe).
7. If you are being asked to test code or run specific commands or scripts, provide the evidence of your outputs through a screenshot and/or screen video-recordings and copy the same into the document.
8. Upon completion, download a .PDF version of the google doc document and submit the same along with all the supplementary files (videos, pictures, scripts etc).

Full Name: Tracy Michaels
Campus ID: tmichaels1
Panther #: 002430918

1)

midterm2 > C question1.c

```c
#include<stdio.h>

//Tracy Michaels
//This program sorts an array and displays it in either
//ascending or descending order
//based on user input

void sort_numeric(float[], int);
void display(float[], int);
void swapFloats(float *, float *);

int main(){

    float array[] = {10, 0.25, 2342, 3.145435, 6, 6, 5.999, 2, 5, 109.56};
    int size = sizeof array/ sizeof array[0];

    sort_numeric(array, size);
    display(array, size);
    return 0;
}

//performs a bubble sort algorithm on array
void sort_numeric(float arr[], int num){
    int i,j;

    for(i = 0; i < num - 1; i++){
        for(j = i+1; j<num; j++){
            if(arr[i] > arr[j]) {
                float temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
}

//displays array in order according to user input
void display(float a[], int n){
    char input[20];
    int index;
    printf("Sort by Ascending(A) or Descending(D): ");
    scanf("%s", input);

    switch(input[0]){
        case 'A':
        case 'a':
            for(index = 0; index < n; index++){
                printf("%f\n", a[index]);
            }
            break;
        case 'D':
        case 'd':
            for(index = 0; index < n; index++){
                printf("%f\n", a[n - 1 - index]);
            }
            break;
        default:
            printf("Invalid input");
    }
    printf("\n");

}
```

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ gcc -o question1 question1.c
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question1
Sort by Ascending(A) or Descending(D): a
0.250000
2.000000
3.145435
5.000000
5.999000
6.000000
6.000000
10.000000
109.559998
2342.000000

[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question1
Sort by Ascending(A) or Descending(D): d
2342.000000
109.559998
10.000000
6.000000
6.000000
5.999000
5.000000
3.145435
2.000000
0.250000
```

2)

```c
#include<stdio.h>
#include<string.h>

//Tracy Michaels
//This program sorts an array of strings and displays it in either
//ascending or descending order
//based on user input

void sort_alphabet(char *arr[], int);
void display(char* a[], int);

int main(){
    //initialize array of strings
    char * array[] = {"Systems",
                      "Programming",
                      "Deep",
                      "Learning",
                      "Internet",
                      "Things",
                      "Robotics",
                      "Course"};

    int size = sizeof array/ sizeof array[0];

    sort_alphabet(array, size);
    display(array, size);
    return 0;
}

//performs a bubble sort algorithm on array
void sort_alphabet(char* arr[], int num){

    int i,j;

    char *temp;

    for(i = 0; i < num - 1; i++){
        for(j = i+1; j<num; j++){
            if(strcasecmp(arr[i], arr[j]) >= 0) {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
}

//displays array in order according to user input
void display(char* a[], int n){
    char input[1];
    int index;
    printf("Sort by Ascending(A) or Descending(D): ");
    scanf("%s", input);

    switch(input[0]){
        case 'A':
        case 'a':
            for(index = 0; index < n; index++){
                printf("%s\n", a[index]);
            }
            break;
        case 'D':
        case 'd':
            for(index = 0; index < n; index++){
                printf("%s\n", a[n - 1 - index]);
            }
            break;
        default:
            printf("Invalid input");
    }
    printf("\n");

}
```

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question2
Sort by Ascending(A) or Descending(D): a
Course
Deep
Internet
Learning
Programming
Robotics
Systems
Things

[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question2
Sort by Ascending(A) or Descending(D): d
Things
Systems
Robotics
Programming
Learning
Internet
Deep
Course
```

3)

midterm2 > C question3.c

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

//Tracy Michaels
//This program sorts an array of strings given by user
//and stores them dynamically in an array
//and displays it in either
//ascending or descending order
//based on user input

void sort_alphabet(char *arr[], int);
void display(char* a[], int);

int main(){

    char **array = malloc(1);
    int size = 0;
    char input[15];


    //get user input
    printf("Enter words ('q' to quit): \n");
    do{
        scanf("%s", input);
        array = (char **)realloc(array, (size + 1) * sizeof(char *));
        array[size++] = strdup(input);
    } while (strcasecmp(input, "q") != 0);



    //sort array and display
    sort_alphabet(array, size -1);
    display(array, size -1);

    //deallocate memory for each element
    int i = 0;
    for(i = 0; i < size; i++){
        free(array[i]);
    }

    //deallocate array
    free(array);

    return 0;
}

//performs a bubble sort algorithm on array
void sort_alphabet(char* arr[], int num){

    int i,j;

    char *temp;

    for(i = 0; i < num - 1; i++){
        for(j = i+1; j<num; j++){
            if(strcasecmp(arr[i], arr[j]) >= 0) {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
}

//displays array in order according to user input
void display(char* a[], int n){
    char input[1];
    int index;
    printf("Sort by Ascending(A) or Descending(D): ");
    scanf("%s", input);

    switch(input[0]){
        case 'A':
        case 'a':
            for(index = 0; index < n; index++){
                printf("%s\n", a[index]);
            }
            break;
        case 'D':
        case 'd':
            for(index = 0; index < n; index++){
                printf("%s\n", a[n - 1 - index]);
            }
            break;
        default:
            printf("Invalid input");
    }
    printf("\n");

}
```

Trial 1

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ gcc -o question3 question3.c
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question3
Enter words ('q' to quit):
Did
you
ever
hear
the
Tragedy
of
Darth
Plagueis
the
wise?
I
thought
not.
It's
not
a
story
the
Jedi
would
tell
you.
q
Sort by Ascending(A) or Descending(D): a
a
Darth
Did
ever
hear
I
It's
Jedi
not
not.
of
Plagueis
story
tell
the
the
the
thought
Tragedy
wise?
would
you
you.

[tmichaels1@gsuad.gsu.edu@snowball midterm2]$
```

Trial 2

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question3
Enter words ('q' to quit):
SomeBODY
once
told
me
the
world
is
gonna
roll
me
I
ain't
the
sharpest
tool
in
the
shed

she
was
looking
kind
of
dumb
q
Sort by Ascending(A) or Descending(D): d
world
was
tool
told
the
the
the
SomeBODY
shed
she
sharpest
roll
once
of
me
me
looking
kind
is
in
I
gonna
dumb
ain't

[tmichaels1@gsuad.gsu.edu@snowball midterm2]$
```

Trial 3

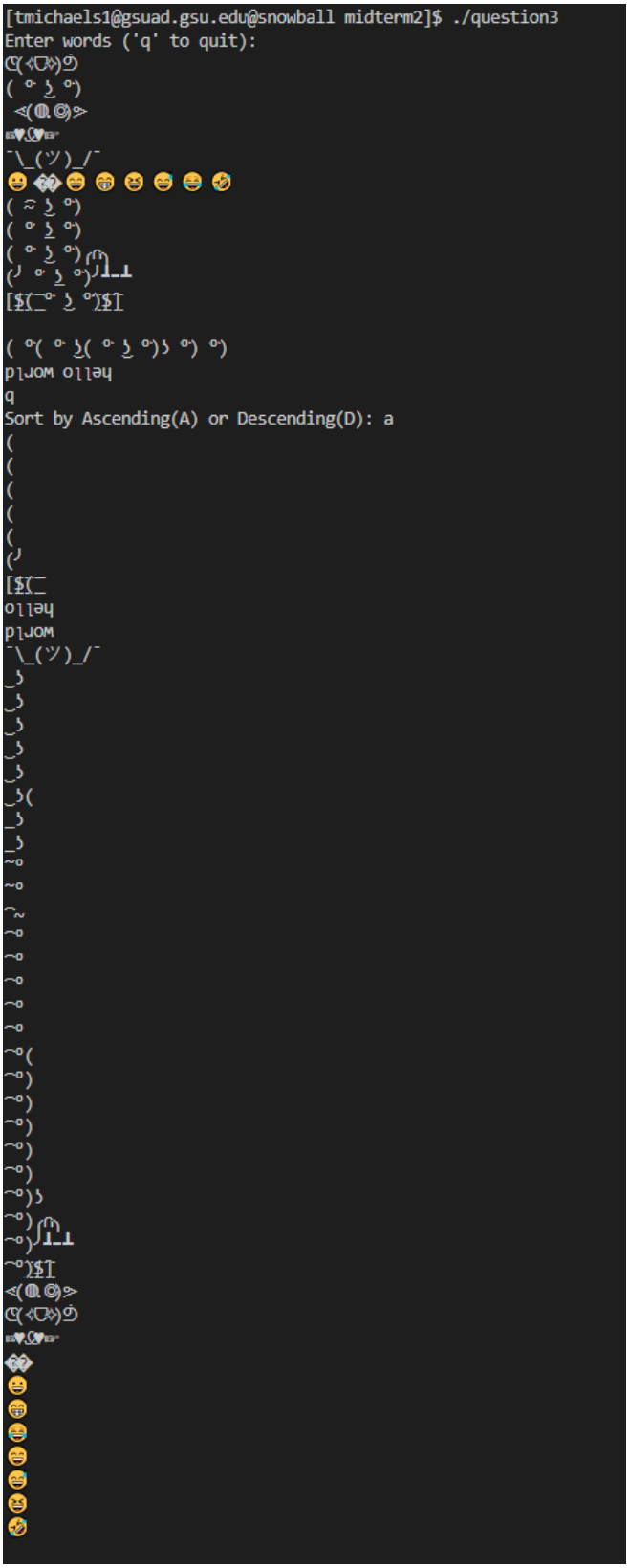
```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question3
Enter words ('q' to quit):
According to all known laws
of aviation,


there is no way a bee
should be able to fly.


Its wings are too small to get
its fat little body off the ground.
q
Sort by Ascending(A) or Descending(D): d
wings
way
too
to
to
to
there
the
small
should
off
of
no
little
laws
known
Its
its
is
ground.
get
fly.
fat
body
bee
be
aviation,
are
all
According
able
a

[tmichaels1@gsuad.gsu.edu@snowball midterm2]$
```

Trial 4

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question3
Enter words ('q' to quit):
In
Japan,
heart
surgeon.
Number
one.
Steady
hand.
One
day
Yakuza
boss
need
new
heart.
I do
operation.
But,
mistake!
yakuza
boss
die!
Yakuza
very
mad.
I
Hide
in
fishing
boat,
come
to
America.
No
english,
no
food
no
money.
Darryl
give
me
job.
now
I
have
house
american
car,
and
new
woman.
Darryl
save
life.
my
big
secret:
I kill
yakuxa
boss
on
purpose.
I good
surgeon.
The
best!
q
Sort by Ascending(A) or Descending(D): d
```

```
q
Sort by Ascending(A) or Descending(D): d
Yakuza
yakuza
Yakuza
yakuxa
woman.
very
to
The
surgeon.
surgeon.
Steady
secret:
save
purpose.
operation.
one.
One
on
Number
now
No
no
no
new
new
need
my
money.
mistake!
me
mad.
life.
kill
job.
Japan,
In
in
I
I
I
I
I
house
Hide
heart.
heart
have
hand.
good
give
food
fishing
english,
do
die!
day
Darryl
Darryl
come
car,
But,
boss
boss
boss
boat,
big
best!
and
american
America.
```

Trial 5


[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question3
Enter words ('q' to quit):
ᕦ(ò_óˇ)ᕤ
(͡° ͜ʖ ͡°)
≼(◉ ◎)≽
ε=♥&♥=з
¯_(ツ)_/¯
😃 😁 😆 😅 😂 🤣
(≈ ͜ʖ ͡°)
(͡° ͜ʖ ͡°)
(͡° ͜ʖ ͡°)ᕗ
(╯ ͡° ͜ʖ ͡°)╯┻━┻
[$(‾ ͡° ͜ʖ ͡°)$]
(͡°(͡° ͜ʖ(͡° ͜ʖ ͡°)ʖ ͡°) ͡°)
dlɹow ollǝɥ
q
Sort by Ascending(A) or Descending(D): a
(
(
(
(
(
(╯
[$(‾
ollǝɥ
dlɹow
¯_(ツ)_/¯
‿ʖ
‿ʖ
‿ʖ
‿ʖ
‿ʖ
‿ʖ(
‿ʖ
‿ʖ
~ᵒ
~ᵒ
˜~
~ᵒ
~ᵒ
~ᵒ
~ᵒ
~ᵒ
~ᵒ(
~ᵒ)
~ᵒ)
~ᵒ)
~ᵒ)
~ᵒ)
~ᵒ)ʖ
~ᵒ)ᕗ
~ᵒ)╯┻━┻
~ᵒ)$]
≼(◉ ◎)≽
ᕦ(ò_óˇ)ᕤ
ε=♥&♥=з
😃
😁
😅
😆
😂
😆
🤣

4)

midterm2 > C question4.c

```c
#include<stdio.h>
#include<string.h>

//Tracy Michaels
//This program registers patients for the covid cavvine
//and displays relavent information about the patients


//struct for patient
struct Patient{
    char first_name[20];
    char last_name[20];
    char birth_date[10];
    char age[2];
    char sex[7];
    char dose_num[1];
    char previous_dose[10];
    char vacc_type[15];
    char zip[5];
    char patientID[9];
};

void _register(struct Patient);
void generate_code(struct Patient);
void retrive(struct Patient);
void display_prompt();


int main(){

    int input;
    struct Patient patients[20];
    int num_patients = 0;
    char *idIn;

    //main logic loop
    do{
        display_prompt();
        scanf("%d", &input);

        switch(input){
            case 1:
                if(num_patients >= 20) {
                    printf("Patient roster full");
                    break;
                }
                _register(patients[num_patients]);
                num_patients++;
                break;
            case 2:
                printf("Type patient id: ");
                scanf("%s", idIn);
                int i;
                for(i = 0; i < num_patients; i++){
                    if(strcasecmp(idIn, patients[i].patientID) == 0) {
                        retrive(patients[i]);
                    }
                }
                printf("Invalid Id\n");
                break;
            case 3:
                break;
            default:
                printf("Invalid Entry\n");
                break;
        }
    } while(input != 3);


    return 0;
}

//diaplays a prompt to the user to start registration process
void display_prompt(){
    printf("\n*********Covid Vaccine Registration**********\n\n");
    printf("(1) Register a Patient\n");
    printf("(2) Display Patient Info\n");
    printf("(3) Quit\n");
    printf("Type number of selection: ");
}

//registers patient and saves infor in a struct
void _register(struct Patient patient){
    char in[20];
    printf("First Name: ");
    scanf("%20s", &in);
    strcpy(patient.first_name, in);
    printf("Last Name: ");
    scanf("%20s", &in);
    strcpy(patient.last_name, in);
    printf("Date of Birth (mm/dd/yyyy): ");
    scanf("%10s", &in);
    strcpy(patient.birth_date, in);
```

midterm2 > C question4.c

```c
                printf("Type patient id: ");
                scanf("%s", idIn);
                int i;
                for(i = 0; i < num_patients; i++){
                    if(strcasecmp(idIn, patients[i].patientID) == 0) {
                        retrive(patients[i]);
                    }
                }
                printf("Invalid Id\n");
                break;
            case 3:
                break;
            default:
                printf("Invalid Entry\n");
                break;
        }
    } while(input != 3);


    return 0;
}

//diaplays a prompt to the user to start registration process
void display_prompt(){
    printf("\n*********Covid Vaccine Registration*********\n\n");
    printf("(1) Register a Patient\n");
    printf("(2) Display Patient Info\n");
    printf("(3) Quit\n");
    printf("Type number of selection: ");
}

//registers patient and saves infor in a struct
void _register(struct Patient patient){
    char in[20];
    printf("First Name: ");
    scanf("%20s", &in);
    strcpy(patient.first_name, in);
    printf("Last Name: ");
    scanf("%20s", &in);
    strcpy(patient.last_name, in);
    printf("Date of Birth (mm/dd/yyyy): ");
    scanf("%10s", &in);
    strcpy(patient.birth_date, in);
    printf("Age: ");
    scanf("%2s", &in);
    strcpy(patient.age, in);
    printf("Sex (Male/Female/Other): ");
    scanf("%7s", &in);
    strcpy(patient.sex, in);
    printf("Dose number (1 or 2): ");
    scanf("%1s", &in);
    strcpy(patient.dose_num, in);
    printf("Previous dose (mm/dd/yyy) or NA: ");
    scanf("%10s", &in);
    strcpy(patient.previous_dose, in);
    printf("Vaccine type (Pfizer/Moderna/Johnson&Johnson): ");
    scanf("%15s", &in);
    strcpy(patient.vacc_type, in);
    printf("Zip code: ");
    scanf("%5s", &in);
    strcpy(patient.zip, in);
    generate_code(patient);
}

//generate unique code for each patient based on patient info
void generate_code(struct Patient patient){

    patient.patientID[0] = patient.first_name[0];
    patient.patientID[1] = patient.last_name[0];
    patient.patientID[2] = patient.age[0];
    patient.patientID[3] = patient.age[1];
    patient.patientID[4] = patient.vacc_type[0];
    patient.patientID[5] = patient.zip[2];
    patient.patientID[6] = patient.zip[3];
    patient.patientID[7] = patient.zip[4];
    patient.patientID[8] = '\0';
    printf("PatientID %s\n", patient.patientID);
}

//displays patient information
void retrive(struct Patient patient){
    printf("\nFirst Name: %s\n", patient.first_name);
    printf("Last Name: %s\n", patient.last_name);
    printf("Date of Birth (mm/dd/yyyy): %s\n", patient.birth_date);
    printf("Age: %s\n", patient.age);
    printf("Sex: %s", patient.sex);
    printf("Dose number: %s\n", patient.dose_num);
    printf("Previous dose (mm/dd/yyy): %s\n", patient.previous_dose);
    printf("Vaccine type: %s\n", patient.vacc_type);
    printf("Zip code: %s\n", patient.zip);
    printf("Patient ID: %s", patient.patientID);

}
```

```
[tmichaels1@gsuad.gsu.edu@snowball midterm2]$ ./question4

*********Covid Vaccine Registration**********

(1) Register a Patient
(2) Display Patient Info
(3) Quit
Type number of selection: 1
First Name: Tracy
Last Name: Michaels
Date of Birth (mm/dd/yyyy): 05/28/1991
Age: 29
Sex (Male/Female/Other): Male
Dose number (1 or 2): 1
Previous dose (mm/dd/yyy) or NA: NA
Vaccine type (Pfizer/Moderna/Johnson&Johnson): Moderna
Zip code: 30518
PatientID TM29M518

*********Covid Vaccine Registration**********

(1) Register a Patient
(2) Display Patient Info
(3) Quit
Type number of selection: 1
First Name: Terry
Last Name: Michaels
Date of Birth (mm/dd/yyyy): 05/29/1954
Age: 66
Sex (Male/Female/Other): Male
Dose number (1 or 2): 1
Previous dose (mm/dd/yyy) or NA: NA
Vaccine type (Pfizer/Moderna/Johnson&Johnson): Pfizer
Zip code: 30107
PatientID TM66P107

*********Covid Vaccine Registration**********

(1) Register a Patient
(2) Display Patient Info
(3) Quit
Type number of selection: 1
First Name: Randi
Last Name: Michaels
Date of Birth (mm/dd/yyyy): 04/04/1954
Age: 66
Sex (Male/Female/Other): Female
Dose number (1 or 2): 1
Previous dose (mm/dd/yyy) or NA: NA
Vaccine type (Pfizer/Moderna/Johnson&Johnson): Johnson&Johnson
Zip code: 30107
PatientID RM66J107

*********Covid Vaccine Registration**********

(1) Register a Patient
(2) Display Patient Info
(3) Quit
Type number of selection:
```