CS 2263 Systems Software Development Lab #06 2022-10-20

> Do not copy and paste the code. All the submissions in the lab have to be created manually. The paper version (if distributed) should be returned in class with the name and the student ID. The final result should be uploaded on D2L using lab06.txt.

Name: _______________ Student ID: _______________ Class: _______________

Instructor: Jong-Kyou Kim, PhD ___________________________

1. Answer the following questions.

   (a) The following code defines the new data type `vector2` and its pointer type using `typedef`. Submit the source code with the name `vector2.h`

   ```
   struct s_vector2 {
     double x, y;
   };

   typedef struct s_vector2 vector2;
   typedef vector2* pvector2;

   void vector2_init(pvector2 this, double x, double y);
   void vector2_copy(pvector2 this, pvector2 other);
   int vector2_add(pvector2 this, pvector2 v1, pvector2 v2);
   int vector2_sub(pvector2 this, pvector2 v1, pvector2 v2);
   const char* vector2_snprint(pvector2 this, char buf[], int bufsiz);
   ```

   (b) The following source code implements initialization and the conversion to string data type of `vector2` type.

   ```
   void vector2_init(pvector2 this, double x, double y)
   {
     this->x = x;
     this->y = y;
   }

   void vector2_copy(pvector2 this, pvector2 other)
   {
     this->x = other->x;
     this->y = other->y;
   }

   int vector2_add(pvector2 this, pvector2 v1, pvector2 v2)
   {
     this->x = v1->x + v2->x;
     this->y = v1->y + v2->y;
     return 0;
   ```

```
}

const char* vector2_snprint(pvector2 this, char buf[], int bufsiz)
{
  snprintf(buf, bufsiz, "(%lf, %lf)", this->x, this->y);
  return buf;
}
```

For example, the following program shows an example use.

```
#include <stdio.h>
#include "vector2.h"

void main() {
  char buf[BUFSIZ];
  vector2 v1;
  vector2_init(&v1, 3, 4);
  vector2_snprint(&v1, buf, BUFSIZ);
  printf("v1 = %s\n", buf);
}
```

The following is the output of the above test program.

```
v1 = (3.000000, 4.000000)
```

Implement the rest of functions and submit the source code `vector2.c`

2. Answer the following questions.

   (a) Create the following text file and submit the file with `makefile`

   ```
   test: test.o vector2.o
           gcc -o test test.o vector2.o
   test.o: test.c vector2.h
           gcc -c test.c
   vector2.o: vector2.c vector2.h
           gcc -c vector2.c
   clean:
           @echo "delete unnecessary files"
           rm -f a.out *.o *~
   ```

   (b) Execute the following command. Submit the output.

   ```
   $ make clean
   ```

   (c) Write the following program and submit the source code with the name `test.c`

   ```
   #include <stdio.h>
   #include "vector2.h"

   void main() {
     char buf[BUFSIZ];
     vector2 res, v1, v2;
     vector2_init(&v1, 3, 3);
     vector2_snprint(&v1, buf, BUFSIZ);
     printf("v1 = %s\n", buf);

     vector2_copy(&v2, &v1);
     vector2_snprint(&v2, buf, BUFSIZ);
     printf("v2 = %s\n", buf);
   ```

```
    vector2_add(&v1, &v1, &v2);
    vector2_snprint(&v1, buf, BUFSIZ);
    printf("v1 = %s\n", buf);

    vector2_init(&v2, 3, 1);
    vector2_snprint(&v2, buf, BUFSIZ);
    printf("v2 = %s\n", buf);

    vector2_sub(&res, &v1, &v2);
    vector2_snprint(&res, buf, BUFSIZ);
    printf("res = %s\n", buf);
  }
```

(d) Execute the following command. Submit the output.

```
$ make test
```

(e) Execute the following command. Submit the output.

```
$ ./test
```